Homecomputer

2 DEUTSCHLANDS ERSTE HEIMCOMPUTER-ZEITSCHRIFT

3. Jahrgang

Doppelausgabe Feb./März '85, 5,50 DM, 48 öS, 6,00 sfr

In dieser Ausgabe:

wieder phantastische

Spielprogramme

und

Anwenderprogramme

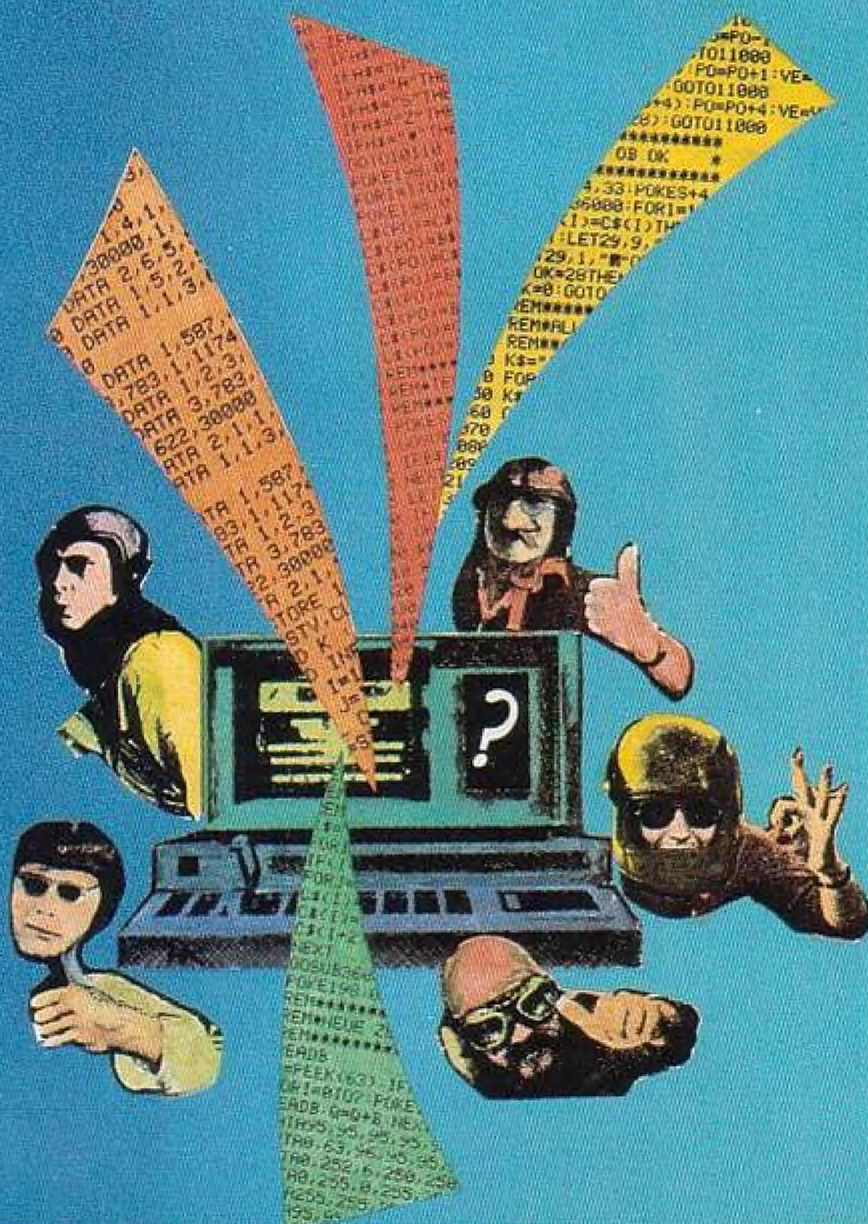
Programm des Monats:

Caverns of Death

Infos
Bücherkiste
Reviews
Tips & Tricks

Alles für:

Commodore 64
VC-20
TI-99/4 A
Apple II
Atari
ZX-Spectrum
ZX-81
Sharp MZ-700
Laser 2001
Dragon 32



Spitzen-Software

Lieber Leser!

Sie haben über Ihren Zeitschriftenhändler die zweite, von der Redaktion des TRONIC-VERLAGS überarbeitete Ausgabe von Homecomputer erhalten. Homecomputer wird weiterhin regelmäßig brandneue Informationen aus der Mikroelektronik bringen, tolle Listings von aktuellen Programmen abdrucken und die populärsten Heimcomputer, wie Commodore 64, VC-20, Dragon 32, TI-99, Apple II, ZX-Spectrum, ZX-81, Laser, Sharp u. a., berücksichtigen.
Auf der Computerszene werden wir flexibel bleiben. Heimcomputer werden mittlerweile immer schneller, immer leistungsfähiger. Viele hoffnungsvolle Newcomer dieser Kategorie schicken sich an, dem Commodore 64 beispielsweise Konkurrenz zu machen. Wir bleiben am Ball. Auch neue Systeme wollen wir berücksichtigen. Homecomputer soll also weiterhin ein Magazin für Anfänger und Fortgeschrittene, Jugendliche und Erwachsene sein. Für alle Mikrocomputer-Interessierten soll Homecomputer spannend, nützlich, unterhaltend und leicht verständlich bleiben.
Interessierte Einsteiger können mit „Homecomputer" in die Welt der faszinierenden Computerspiele einsteigen. Viele Leute verwenden Heimcomputer, um sich mit den beliebten „Arcadespielen" wie Pac-Man oder Blue Max die Zeit zu vertreiben. Mit Hilfe von Homecomputer können Sie noch weiter in die Spielszene eindringen; zudem bieten wir unseren bisher so erfolgreichen (weil schnellen) Software-Service an. Anwenderprogramme, Hilfsprogramme, Simulations- und Abenteuerspiele werden Sie in eine phantastische Welt versetzen. In jeder Ausgabe halten wir einen neuen Bestell-Katalog für Sie bereit. Wir wissen, die richtige Software auszusuchen, kann genauso schwierig und zeitraubend sein wie die Auswahl des „richtigen" Computers. Wir sind Ihnen dabei gern behilflich.

Die Redaktion

INHALT

Letzte Meldung!!!

Unsere Zeitschrift „Homecomputer 2" ist eine Doppelausgabe. Wir haben uns entschlossen, vorübergehend jeden 2. Monat zu erscheinen. Ihr Plus dabei: Wir bringen ab der nächsten Ausgabe mehr Seiten – und noch mehr Listings. Unsere recht kurzfristige Entscheidung, dies so zu machen, werden wir im nächsten Homecomputer ausführlich erklären.

Die Redaktion

Homecomputer

Heft Februar 1985 – Jahrgang 3

Top-Programm des Monats

Blue Bugs



TI-99

Caverns of Death



Commodore 64

Atlantis

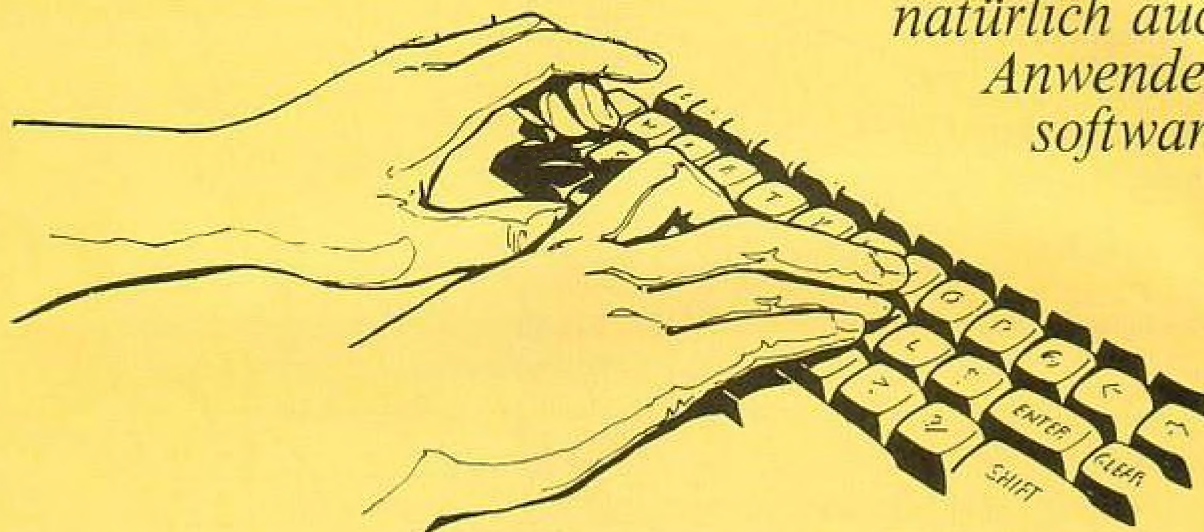


Atari

Hört, hört ...

Wenn Sie in der Lage sind, auch andere an Ihren Phantasien teilhaben zu lassen, versuchen Sie es doch einfach mal mit selbsterstellten Programmen!

Wir suchen ständig neue, phantastische Abenteuer, spannende und lustige Programme und natürlich auch Anwender-software.



Wir brauchen von Ihnen:

Ausführliche Spielbeschreibung
Datenträger (Kassette oder Diskette)
Komplettes Listing (nicht unbedingt erforderlich)

Wir zahlen Ihnen:

DM 120,- pro abgedruckte volle Seite

All dies schicken Sie bitte an den Tronic-Verlag, Postfach, 3444 Wehretal 1

Computercamp Ferienzentrum Schloß Dankern greift den Trend zu LOGO auf

LOGO, die Computersprache „mit der Schildkröte", ist die in den USA offiziell anerkannte und in öffentlichen Schulen gelehrte Computersprache für die Jüngsten. Nachdem bereits in diesem Jahr erkennbar war, daß immer mehr jüngere Kinder an die Terminals drängten, war es da eigentlich nur „logo", auch diese, wegen ihrer Einfachheit besonders für Kinder unter 10 geeignete Programmiersprache mit in unser Unterrichtsprogramm aufzunehmen. Daß gerade im neuen LOGO-Kurs besonderer Wert auf spielerischen „Unterricht" gelegt wird, versteht sich da eigentlich schon fast von selbst. Toll ist, daß durch den neuen LOGO-Kurs auch die jüngsten Computerfans die Möglichkeit bekommen, „fast wie die Großen" zu programmieren und lustige Mal- und Spielprogramme zu gestalten.

Ein neues Unterrichtssystem im Computercamp Ferienzentrum Schloß Dankern

Um den Einzelwünschen der Teilnehmer noch besser gerecht zu werden, ist das gesamte Unterrichtssystem neu aufgebaut worden. Schon bei der Buchung geben die Teilnehmer an, für welchen Kurs bzw. welche Kurse sie sich interessieren. Die Einzelkurse – LOGO, BASIC 1, BASIC 2 und MASCHINENSPRACHE – sind so ausgelegt, daß sie aufeinander aufbauen und daher einzeln oder direkt aufeinander folgend belegt werden können.

Andererseits bleibt aber durch den besonders individuellen Unterricht die Möglichkeit, auch mehrere Wochen hintereinander z. B. den MASCHINENSPRACHEKURS zu besuchen, falls jemand gerade besonderes Interesse hat.

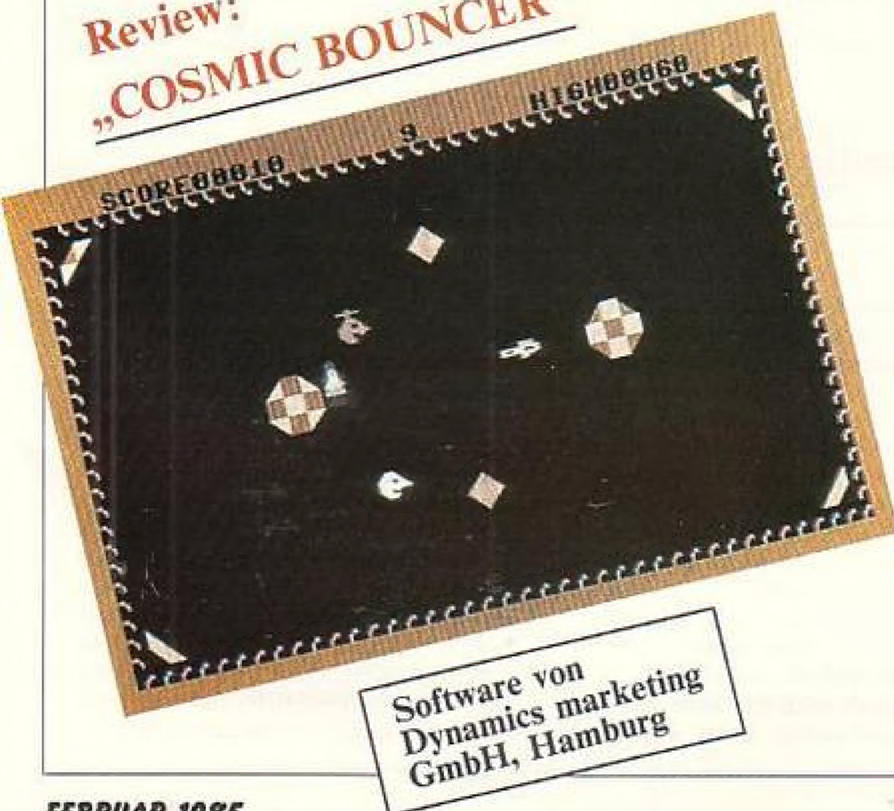
Das Computercamp-Freizeitprogramm 1985 wird noch weiter verbessert

Für alle diesjährigen Teilnehmer wird das zwar unwahrscheinlich klingen, aber trotzdem haben wir noch einige ganz neue Programmpunkte unserer fast unendlichen Liste an Freizeitaktivitäten hinzugefügt. Hier einige der Highlights im Telegrammstil: Schwimmen im Wellenhallenbad Haren ... Video zum Selbermachen ... Ausflüge nach Holland und in den Naturpark Hilterberg ... Besichtigungstouren von Sehenswürdigkeiten wie einer alten Mühle und historischen Ems-Booten ... Eine Schiffstour auf der Ems mit der „Amisia" ... Besuch eines typischen Emsland-Bauernhofes ... und viele Wettbewerbe, von Tischtennis bis zum Puddingwettessen.

Für weitere Informationen wendet man sich an:
Buchungsbüro Computercamp Ferienzentrum Schloß Dankern, Holztwiete 4 d, 2000 Hamburg 52, Telefon 0 40 / 82 79 42.
Ab 14. 1. 1985: Goßlerstraße 21, 2000 Hamburg 55, Telefon 0 40 / 86 23 44.

* * * * * * * * * * * * * * * *

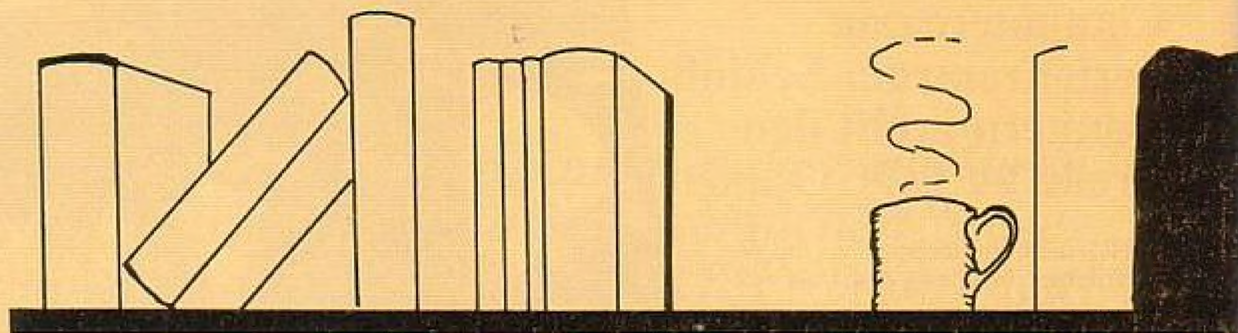
Review: „COSMIC BOUNCER"



Software von Dynamics marketing GmbH, Hamburg

Commodore 64

Zwischen zwei Planeten pendelt eine Raumfähre, die von feindlichen Gestalten angegriffen wird. Mit Ihrem Raumschiff sichern Sie den Weg der Weltraumfähre. Dazu steht Ihnen ein Energieball zur Verfügung, der aber immer wieder aufgefangen werden muß, bevor ein Angreifer damit unschädlich gemacht werden kann. Sie haben insgesamt neun Schiffe für Ihre Mission. Manchmal müssen Sie ein Schiff opfern, um die Zerstörung der Raumfähre zu vermeiden. *Ein Programm auf Kassette, empf. Verbraucherpreis: 19,95 DM*



Homecomputer-Bücherkiste

Was Sie schon immer über Computer wissen wollten von Daniel Cohen



Es ist an der Zeit, mehr über Computer zu wissen. Computer – vom Mikro – bis zum Großrechner – betreffen inzwischen uns alle, jeden Tag, Privat und beruflich, selbst in Situationen, in denen ein Computer eigentlich gar nichts zu suchen hat. Durch dieses Buch werden Sie in Form eines Dialogs mit dem Computer vertraut gemacht. Es stellt klare Fragen – Ihre Fragen! Es gibt einfache Antworten – Antworten eines Experten! So erfahren Sie alles über die Geschichte und Entwicklung der Computer, die Technologie und Terminologie, die Arbeitsweise und den Einsatz, die Programmierung und die Anwendung usw. Dieses Computerbuch beantwortet Ihnen über 200 Fragen zum Thema Computer, auch Fragen, die Sie sich noch gar nicht gestellt haben. Das wichtigste Kriterium ist es, daß Sie sich Zeit für dieses Buch und seine Themen lassen. Es lohnt sich!

Heyne Computer Bücher Nr. 15/8, DM 7,80, Originalausgabe

Testen Sie Ihre Computer-Intelligenz von Alfred W. Munzert

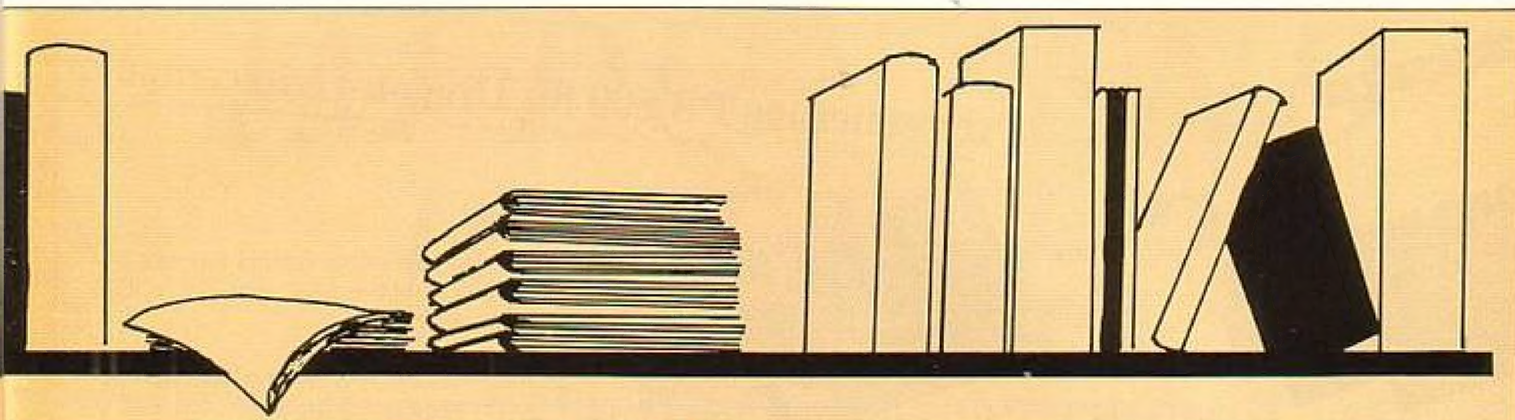


Ein spannendes, amüsantes Testprogramm – der spielend leichte Einstieg in die faszinierende Welt der Mikrocomputer.

Der Autor hat es verstanden, auf einfache, verständliche, fast spielerische Weise eine Einführung in Logik und Technik von Computern zu geben. Er beantwortet Fragen wie:
- Was ist ein Computer?
- Wie funktioniert das Gerät?
- Welchen Nutzen bringt mir der Heimcomputer?

Daneben wird auch eine Serie von sorgfältig ausgewählten Tests angeboten, die Ihnen mit Witz und Spannung zeigt, wie hoch Ihr persönlicher Computer-IQ ist. Er beantwortet Ihnen auch Fragen ob Sie besonders begabt für den Computer sind, ob Sie der geborene Programmierer sind oder worin Ihre besondere Stärke liegt.
Dieses Buch ist die ideale Vorbereitung für den Umgang mit Mikrocomputern. Für den Fortgeschrittenen stellt es eine Herausforderung an seine logischen Fähigkeiten dar.

Heyne Computer Bücher Nr. 15/1, DM 9,80, Originalausgabe



Commodore 64 — Michael Boom

Der Commodore 64 gehört zur neuen Generation von Computern, die sich durch hohe Leistungsfähigkeit und große Speicherkapazität auszeichnen. Sein niedriger Preis macht ihn für einen Benutzerkreis interessant, für den der Erwerb eines Computers aufgrund der damit verbundenen Aufwendungen bislang nicht in Frage gekommen ist. Mit dem Commodore 64 können Sie Ihr Konto oder die Bücher eines kleineren Geschäftes führen, Briefe an Freunde schreiben, Schach spielen oder Raumschiffe durch feindliches Gebiet im Weltall steuern.
Das Taschenbuch aus der Reihe *»Goldmann computer compact«* gibt Ihnen eine Einführung in die Handhabung und Arbeitsweise des Commodore 64 und bietet einen Überblick über die Programme und Produkte, mit denen Sie das System erweitern können.
Außerdem finden Sie eine ausführliche Darstellung des Commodore 64-Systems, seiner einzelnen Elemente und ihrer Aufgaben. *Computer Compact, Goldmann Verlag, München.*
Wenn Sie sofort mit Ihrem Computer arbeiten wollen, lesen Sie Kapitel 3 und springen dann zum Anhang, der Sie in das Programmieren und Laden von Programmen einweist. Wenn Sie aber etwas Geduld haben, wird Sie die nähere Betrachtung des C 64-Computersystems darin unterstützen, Ihr Gerät besser zu verstehen und zu kontrollieren.

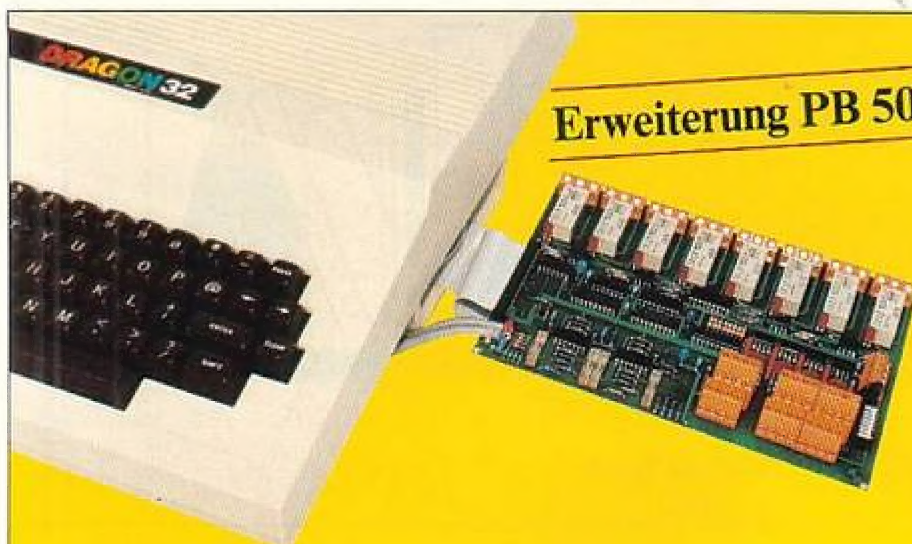


Basic — Richard G. Peddicord

Sie haben sich entschlossen, sich mit der Computer-Programmiersprache BASIC vertraut zu machen. BASIC, die Abkürzung steht für Beginner's Allpurpose Symbolic Instruction Code und bezeichnet die am leichtesten erlernbare höhere Programmiersprache, ist eine Sprache wie jede andere auch, aber viel einfacher. Um mit BASIC umgehen zu können, müssen Sie einige englische Ausdrücke lernen, die allgemeinverständlich scheinen, in BASIC aber einen eindeutigen Sinn haben. Sie müssen genau wissen, wann und wie jedes Wort zu benutzen ist; denn nur so können Sie einem Computer korrekte Anweisungen geben.
Einem Computer Instruktionen zu geben, heißt, ihn zu programmieren. Das Programm sagt dem Computer genau, was in welcher Reihenfolge zu tun ist. Jede Aufgabe, wie einfach sie auch sein mag, muß dem Computer durch ein Programm zugewiesen werden, sonst kann er sie nicht ausführen. Im übrigen befolgen Computer die Programmanweisungen präzise, d. h., sie übernehmen auch alle Fehler des Programmierers und führen sie aus. Korrektes Programmieren verlangt daher sorgfältige Beachtung aller Details.
Was ist ein Computer? Er ist eine Maschine, die Ihnen hilft, Probleme beim Erfassen und Verarbeiten von Informationen zu bewältigen. Mit dem Computer läßt sich ein Großteil der Papierarbeit im Büro erledigen. Sie können ihm Buchhaltung und Finanzverwaltung Ihrer Firma anvertrauen, er führt komplizierte Berechnungen durch, bleibt wichtigen Vorgängen auf der Spur und bietet Ihnen sogar die Möglichkeit zu unterhaltsamen, anspruchsvollen Spielen.



Erweiterung PB 500 für Dragon-Heimcomputer



Die Fa. appli-data GmbH hat eine Erweiterungsplatine für den DRAGON entwickelt.

Die heißt PB 500, ist eine leistungsstarke Erweiterungsplatine für den DRAGON und bietet folgende Leistungsmerkmale:

- direkter Anschluß an das BUS-Erweiterungsport
- Digitaleingänge mit Schutzbeschaltung und Eingangsfilter (Industriestandard)
- Simulationsschalter für jeden Digital-Eingang
- Digital-Relais-Ausgänge 220 V – 8 A/2000 VA
- Optokoppler und Anzeige LEDs für alle Digital-Ein-/Ausgänge
- 4 Analog-Eingänge für direkte Temperaturmessung
- keine Einstellung erforderlich: alle Kanäle sind geeicht
- festeingestellte Temperaturbereiche –20 °C bis +43 °C und 10 °C bis 73 °C
- alle Anschlüsse über Schraubklemmen
- sehr einfache Handhabung der Hardware und Software

Der anwenderfreundliche und praxisgerechte Aufbau erschließt neue Anwendungsmöglichkeiten, z. B. Alarm- und Störmeldesysteme, Hausleittechnik, Datenerfassung, intelligentes Steuerwerk für Heim und Hobby (Heizungssteuerung, Modellbahnsteuerung ...).

Schnittstellenumschalter für Centronics-Schnittstellen

Neu auf dem Markt sind seit Oktober 84 die Schnittstellenumschalter DUO-PER C und DUO-PER CI für Centronics-Schnittstellen. Der Umschalter verfügt über einen Computeranschluß und 2 Ausgänge für Drucker oder ähnliche Peripheriegeräte. Die Umschaltung erfolgt über einen Schalter an der Frontplatte, der die interne Elektronik ansteuert. Die Schnittstellensignale werden intern durch eine Elektronik regeneriert, damit sie störungsfrei beim Peripheriegerät ankommen.



Das Gerät gibt es in drei Ausführungen. Der DUO-PER C ist mit externer und interner Stromversorgung erhältlich. Die interne Stromversorgung erfolgt über ein Steckernetzteil. Bei Externer Stromversorgung muß die 5 V-Versorgung über eines der Schnittstellenkabel aus den angeschlossenen Geräten erfolgen. Der DUO-PER CI ist die industrielle Ausführung in einem Metallgehäuse. Die Netzversorgung ist eingebaut. Alle Geräte werden über Stecker an der Rückseite angeschlossen. Der eingeschaltete Kanal wird durch eine Leuchtdiode an der Frontplatte angezeigt.
Nähere Informationen: BAUZ GmbH, Postfach 13 29, 2730 Zeven

Ein Nadel-Matrixdrucker von Olympia für jeden Mikrocomputer

Von Olympia gibt es jetzt aus der compact-Serie auch einen Matrixdrucker, den Olympia electronic compact NP. Daneben gibt es die bereits bekannte Schreibmaschine mit werksseitigem Interface, electronic compact 2, sowie einen Typenraddrucker ohne Tastatur, die Maschine electronic compact RO.

- Per Tastendruck kann in Schönschrift gedruckt werden. Dazu wird jede Zeile zweimal mit leichtem Versatz gedruckt.
- Durch eine Gummiwalze wird ein sehr leiser Druck erzielt.
- Mit den Bedientasten kann der linke und der rechte Rand gesetzt werden.
- Der Traktor ist auch auf schmales Papier (einbahnige Etiketten) einstellbar.

Der Preis beträgt für die Ausführung mit Centronics-Interface DM 1698,– und für Centronics und V.24-Interface (alle Baudraten, Hardware- und Softwareprotokoll) DM 1948,– inkl. MwSt. Eine Ausführung für den C64 mit Grafikfähigkeit kostet DM 1948,–. Info.: iti-Datentechnik, Telemannstraße 18, 7250 Leonberg, Tel.: 0 71 52 / 63 05 o. 7 10 74

IM DUTZEND BILLIGER

ist auch Ihre Homecomputerzeitschrift.

und zwar um bis zu 25 %,
wenn Sie sich für ein Jahresabonnement
von
HOMECOMPUTER, CPU
oder HC & CPU als Kombination
entscheiden

Statt 66,- DM für 12 Hefte HC oder CPU zahlen Sie im Jahresabonnement nur 55,- DM. Beide Hefte im Abonnement kosten statt 132,- DM nur 100,- DM.

Bei Lieferung in das europäische Ausland beträgt das JAHRESABONNEMENT HC oder CPU 80,- DM (nach außereuropäischem Ausland 110,- DM). Das KOMBINATIONSABO von HC und CPU 150,- DM (europ. Ausland). Übersee 200,- DM.

Verlags-Garantie

★ Sie erhalten HOMECOMPUTER und CPU ab der nächsterreichbaren Ausgabe ★
★ Lieferung erfolgt frei Haus inkl. Mehrwertsteuer ★
★ Die Zustellgebühren sind im günstigen Abonnementspreis bereits enthalten ★
★ Es entstehen Ihnen keine weiteren Kosten ★

Ausschneiden und einsenden an:

Verlagsunion

Friedrich-Bergius-Straße 20
Postfach 57 07

6200 Wiesbaden

Bestellgutschein

Ja, ich möchte ins Computerleben einsteigen und bestelle deshalb:

☐ Jahresabonnement von Homecomputer (12 Hefte)
☐ Jahresabonnement von CPU (12 Hefte)
☐ Kombi-Abonnement von HC + CPU (24 Hefte)

Name/Vorname

Straße, Nr. | Plz, Ort

Ich wünsche folgende Zahlungsweise

☐ Bargeldlos durch Bankeinzug: | Bankleitzahl

Geldinstitut | Konto-Nr.

☐ gegen Rechnung
☐ gegen Vorkasse

Datum, Unterschrift

Postgiroamt Ffm.,
Kt.-Nr. 4192 90-601 BLZ 500 100 60
(Betreffendes bitte ankreuzen)

Abonnements-Kündigungen:
6 Wochen vor Ablauf des Jahres-Abonnements.
Außerdem kann ich diese Bestellung innerhalb von 10 Tagen beim Verlag widerrufen!

Datum, Unterschrift

Erfahrungsbericht – Dragon 64 –

Optischer Eindruck

Farbgebung:

– Die Farbe des Dragon 64 ist ein helles Grau, das sehr gut mit Peripheriegeräten (Monitor, Drucker, Datenrecorder) anderer Hersteller harmoniert, so daß auch bei Benutzung von Geräten, die nicht zum Dragonsystem gehören, der Eindruck einer homogenen Einheit erhalten bleibt.

(Viele Peripheriegeräte anderer Hersteller weisen gleiche Farbgebung auf: Sanyo-Farbmonitor, Seikosha-Drukker, Sanyo-Datarecorder, etc.)

Form:

– Meines Erachtens hat die äußere Gestalt des Dragon, sowohl des 32K-Modells aus auch des 64k-Modells, professionellen Charakter. (Ich denke hier besonders an die Ähnlichkeit mit einem amerikanischen PC aus dem Silicon-Valley, der eine Frucht als Markenzeichen trägt) Zu bemängeln wäre hier allerdings, daß das Gehäuse nicht kräftig genug ist, einen Datenmonitor zu tragen, so daß eine professionelle Konfiguration einigen Platz benötigt!

Anordnung der Tasten:

– Reset-Taste und Ein-/Ausschalter sind versteckt an der Seite, bzw. an der Rückseite des Gerätes angebracht. Diese Anordnung, insbesondere der Reset-Taste, beugt einer ungewollten Betätigung durch den User vor.

Leider besitzt der Dragon keine optische Anzeige über den Betriebszustand. Behelfsmöglichkeit: Benutzung einer Steckdosenleiste mit Schalter. Dies hat außerdem noch die Vorteile

a) Die Peripherie ist an einem Anschluß

b) der Netzstecker braucht bei Nichtbenutzung des Computers nicht gezogen zu werden.

Die Anordnung der Tastatur ist gleich einer modernen elektrischen Schreibmaschine oder einem Fernschreiber, d. h. das Tastenfeld liegt nahezu in einer Ebene – die Tastenreihen sind nur unmerklich voneinander gestuft.

Weiter Seite 11

Sounds aus dem Dragon

Der Dragon-Computer verfügt zur Erzeugung von Tönen über eine „Play"- und eine „Sound"-Anweisung. Diese beiden Befehle sind von Basic aus ansteuerbar. Darüber hinaus besteht jedoch noch eine andere Möglichkeit, dem Dragon Töne zu entlocken.

Die nachfolgenden Beispiele geben einen Einblick darüber, wie es möglich ist, verschiedene Geräuscheffekte etwas anderer Art zu erzeugen.

Dazu vorab eine kurze Beschreibung:

Das Programm spricht die Speicherstellen &HFF23 und &HFF28 im I/O i-Port des Prozessors MC 6809 an. Die MPU steuert hier, einfach ausgedrückt, u. a. die Tonausgabe.

Durch diese Art der Programmierung kann der User jedoch lediglich Soundeffekte durch entsprechende Kontrolle über den I/O-Port, genauer gesagt über die Lautstärke und Schwellungslänge erzeugen.

Doch wie das Beispielprogramm zeigt, sind auch hier interessante Geräuschvariationen möglich.

```
100 FOR I=255 TO 1 STEP-1
110 POKE &HFF23,100
120 POKE &HFF28,I
130 POKE &HFF28,0
140 NEXT I
150 POKE &HFF23,255
160 POKE &HFF28,255
170 GOTO 100
```

Natürlich sind die Variationsmöglichkeiten von Basic aus begrenzt, da schon allein durch die Verarbeitungsgeschwindigkeit schnelle Toneffekte nicht erfolgen können.

Dennoch läßt das obige Basicprogramm einige Experimente zu, z. B. die Änderung der Schleife für I= ..... in andere Schritte oder Zählung.

Um diese Art der Programmierung wesentlich schneller arbeiten zu lassen, ist Maschinensprache unumgänglich.

Das folgende Assemblerlisting wurde mit dem Assembler „Dream" erstellt und bietet mehr Anwendungsmöglichkeiten:

```
7D00                  ORG  32000
7D00  867F            LDA  #127    Tonlänge Werte von 127–200
7D02  C6FF            LDB  #255    Lautstärke Wert <255
7D04  F7FF23    MO1   STB  $FF23
7D07  B7FF28          STA  $FF28
7D0A  4C              INCA
7D0B  F7FF23          STB  $FF23
7D0E  B7FF28          STA  $FF28
7D11  8E00C8          LDX  #200    Dauer der Schwellung
7D14  301F      MO2   LEAC -1,X
7D16  26FC            BNE  MO2
7D18  F7FF28          STB  $FF28
7D1B  81C8            CMPA #200    Tonlänge maximal
7D1D  2DE5            BLT  MO1
7D1F  C6FF            LDB  #255
7D21  F7FF23          STB  $FF23
7D24  F7FF28          STB  $FF28
7D27  39              RTS
```

Das Maschinenprogramm wurde hier ab Speicherstelle 32000 im Ram untergebracht.

Es läßt sich jedoch durch die relative Adressierung überall im Ram verschieben. Diejenigen User, die über den „Dream" verfügen, können das Quellprogramm, so wie oben angegeben ist, direkt mit dem Assembler eingeben und entsprechend ab &H4E21 starten oder für ihre eigenen Anwendungen im Speicher verschieben.

Alle übrigen User müssen das Programm mit Hilfe des Poke-Befehls in den Speicher bringen.

Dazu helfen folgende Befehle:

```
10  FORI=&H7D00 TO &H7D27
20  READA$:POKEI, VAL ("&H"+A$)
30  NEXTI
```

```
40 DATA 86, 7F, C6, FF, F7, FF, 23, B7, FF, 28, 4C, F7, FF, 23, B7, FF, 28, 8E,
   00, C8, 30, 1F, 26, FC, F7, FF, 28, 81, C8, 2D, E5, C6, FF, F7, FF, 23, F7, FF,
   28, 39
```

Damit ist das Programm im Speicher und kann mit EPEX &H7D00 aufgerufen werden. Versuchen Sie einmal folgendes Beispielprogramm:

```
200 I1=127
210 FORI=1TO90
220 I1=I1+1
230 POKE &H7D01,I1
240 EXEC &H7D00
250 NEXTI
```

Wegen der besseren Übersicht wurde hier das Beispielprogramm Zeile für Zeile aufgelistet. In einem Anwenderprogramm empfiehlt es sich, wegen der höheren Arbeitsgeschwindigkeit den Gesamtbefehl in eine Zeile zu setzen.
Dadurch, daß mit Hilfe des Poke-Befehls das Maschinenprogramm mit ständig anderen Werten versehen werden kann, ist eine große Anwendungsvielfalt gegeben.

Um die Geräuscheffekte zu ändern, muß der User lediglich die entsprechenden Parameter angeben:

Beginn der Wiederholungszählung	= &H77ED
Toneffektvariable	= &H77EF
Begrenzung der 1. Tonfolge	= &H77F5
Start der 2. Tonfolge (Anhangeffekt)	= &H77F9
Begrenzung der 2. Tonfolge	= &H77FF
Anzahl der Effektwiederholungen	= &H7804

Die entsprechenden Parameter können durch den Poke-Befehl in einem laufenden Programm in die Maschinenroutine eingestuft werden. So können immer andere effek.
Das Maschinenprogramm kann noch ergänzt werden, wenn der User an die beiden „Rom-Routinen“ ab &H780C noch folgende Routinen anfügt:

```
JSR  $BAD6
JSR  $BAD4
```

Dadurch ergeben sich etwas andere Effekte.
Das Maschinenprogramm kann genauso in den Speicher gebracht werden, wie es bereits im ersten Beispiel beschrieben wurde (Poke-Befehl/Data-Zeilen).
Neben dieser Art, Toneffekte zu erzielen, gibt es eine völlig andere. Hier kann sich der User die entsprechenden Speicherstellen in der Zero-Page und im Rom des Dragon zu Hilfe nehmen.
Die folgende Maschinenroutine wurde ebenfalls mit „Dream“ erstellt und ab Speicherstelle 30700 dissassembliert:

```
77EC  C601          LDB  #0      Wiederholungszähler
77EE  8677     LO2  LDA  #119    Toneffektwert von 1–120
77F0  4C       LO1  INCA         Tonwert um 1 erhöhen
77F1  BD7808        JSR  UP1
77F4  817F          CMPA #127    Begrenzung 1. Tonfolge
77F6  2DF8          BLT  LO1
77F8  86A7          LDA  #167    zweite Tonfolge
77FA  4C       LO3  INCA
77FB  BD7808        JSR  UP1
77FE  81FF          CMPA #255    Begrenzung 2. Tonfolge
7800  2DF8          BLT  LO3
7802  5C            INCB
7803  C10A          CMPB #10     zehnmal Effekt wiederholen
7805  2DE7          BLT  LO2
7807  39            RTS
7808  978C     UP1  STA  $8C     Zero-Page Tonlänge/Höhe
780A  3402          PSHS A
780C  BDBAB3        JSR  $BAB3
780F  BDBAC5        JSR  $BAC5
7812  3502          PULS A
7814  39            RTS
```

Das Programm läßt sich mit EXEC &H77EC aufrufen und bietet eine noch vielseitigere Anwendungsmöglichkeit wie die zuerst beschriebene Art der Tonerzeugung.
Hiermit lassen sich alle möglichen Geräuscheffekte wie Sirene, Dampflok oder anderes erzeugen. Das bleibt ganz der Phantasie des Anwenders überlassen.

Tastatur

Anschlag:
– Die Tastatur des Dragon-Computers erfüllt professionelle Erwartungen. Sie ist leichtgängig, prellt nicht und ist auch für grobe Männerfinger ohne Schwierigkeiten zu bedienen.
Die einzelnen Tasten sind ein-, bzw. zweifach belegt, was die Handhabung wesentlich erleichtert.

Tastatursystem:
– Die Tasten sind nach amerikanischem System angebracht, was für Blindschreiber evtl. zu Problemen führen kann, da „Z“ und „Y“ vertauscht sind – in Bezug auf deutsche Tastaturen!
Für den Heimanwender bedeutet dies wohl kaum einen Nachteil, auch für einen Programmierer ist diese Art der Anordnung nicht problematisch. Kritik daran kann von einem Operator kommen, der an deutsche Tastaturen gewöhnt ist.
Die Enter-Taste ist groß gehalten und günstig am rechten Rand positioniert – neben der Clear-Taste.
Diese Nachbarschaft ist für meinen Geschmack nicht optimal, da eine unbeabsichtigte Betätigung einer der beiden Tasten, besonders dann, wenn eine dieser Tasten innerhalb eines Programmes (oder beide) mit spezifischen Funktionen belegt ist (sind), ungewünschte Reaktionen hervorruft.

Schnittstellen

Anordnung:
– Linke Seite:
Drucker-Interface
RS 232
Kassetten-Recorder
Joysticks – links/rechts –
Video-Ausgang
Rechte Seite:
Steckmodulschacht kombiniert mit Einschubschacht für das Diskettenlaufwerk
Rückseite:
Monitorausgang
Netzversorgung
Zur Vermeidung von Kabelsalat um das Gerät herum wäre eine Anordnung der Schnittstellen an der Rückseite zweckmäßiger. Bei der Benutzung von beiden Joysticks und des Kassettenrecorders liegen 3 Kanal über/unter/nebeneinander. Eine klare Trennung der Joystickschnittstellen und des Kassettenrecordereingangs erscheint mit benutzerfreundlicher. Schlecht finde ich die Lösung, den Schacht für die Steckmodule mit dem Einschubschacht für das Diskettenlaufwerk zu kombinieren – außerdem ist die rechte Seite nicht der beste Platz, ein Diskettenlaufwerk anzuschließen, da hierdurch schon wieder
Weiter Seite 35

BLUE-BUGS

TI 99/4 A

Das Spiel ist auf dem TI 99/4 A in Extended Basic geschrieben und wird mittels Joystick 1 gesteuert.



Weit draußen in einer entfernten Galaxis lebt ein Volk auf dem 2. Mond des Planeten Carylos. Da der Mond keine Atmosphäre hat, leben die Bewohner in kuppelartigen Städten. Eines Tages werden diese friedlichen Bewohner jedoch von blauen, fliegenden Insekten, den „BLUE BUGS" überfallen, welche die Bewohner vernichten wollen. Als einzige Waffe steht eine Torpedokanone zur Verfügung, mit welcher Sie die „BLUE BUGS" vernichten müssen.

Nach dem Eintippen wird das Spiel mit RUN gestartet. Besonderheiten beim Spiel sind nicht zu beachten, da selbst die ‹ALPHA LOCK› Taste eingerastet sein kann. Das Spiel wird Ihnen eine kleine akustische Überraschung servieren. Hierzu sei allerdings nichts gesagt. Lassen Sie sich einfach einmal überraschen.

Nachdem das Titelbild erschienen ist, können Sie mit dem Joystick den Level wählen, in dem Sie beginnen wollen. Der Level (0 bis 9) wird oben links im Bild eingeblendet, und verändert sich, sobald Sie den Joystick bewegen. Haben Sie Ihren Level gewählt, so drücken Sie den Aktionsknopf, um das Spiel zu beginnen.

Die Blue Bugs greifen in Wellen an. Vor jeder Welle wird der augenblickliche Level im Planeten angezeigt. Ein BLUE BUG erscheint mit lautem Getöse auf dem Bildschirm. Dieser BLUE BUG wirft seine Eier ab, die Ihre Laserkanone zerstören können, sobald diese berührt wird. Eine Kanone wird immer dann zerstört, wenn es einem BLUE BUG gelingt bis zum Boden des Mondes vorzudringen, wenn ein BLUE BUG die Kanone berührt, oder wenn diese von den Eiern getroffen wird. Insgesamt stehen 5 Kanonen pro Spiel zur Verfügung.

Nachdem eine Welle vorbei ist, wird der Level erhöht (maximal Level 9), und eine andere Art von BLUE BUGS greift an.

Insgesamt gibt es 3 Arten von BLUE BUGS:

1. Bird BLUE BUGS
2. Spider BLUE BUGS
3. Worm BLUE BUGS

Diese 3 Arten wechseln sich von Welle zu Welle mit ihrem Angriff ab.

Nachdem Sie alle 5 Kanonen verloren haben, erscheint wieder das Titelbild, und Sie können wieder den LEVEL wählen.

Anzahl der BLUE BUGS pro Welle und Punkte pro Treffer:

LEVEL	Anzahl BLUE BUGS	Punkte pro Treffer
0	8	10
1	10	20
2	12	30
3	14	40
4	16	50
5	18	60
6	20	70
7	22	80
8	24	90
9	26	100

Wir wünschen ein Errorfreies Eintippen, und viel Spaß bei ... BLUE BUGS

```
100 !***********************
110 !*  B L U E - B U G S  *
120 !*---------------------*
130 !* TI 99/4 A & X-BASIC *
140 !* Steuerung: Joystick *
150 !*---------------------*
160 !* (c) 1984 Tronicsoft *
170 !* all Rights V.Becker *
180 !***********************
190 !
200 DATA 659,523,659,523,698,587,783,659,659,523,659,523,587,493,523,440,659,523
,659,523,659,523,30000,30000
210 DATA 659,523,659,523,698,587,783,659,659,523,659,523,587,493,523,440,440,349
,440,349,440,349,30000,30000
220 DATA 440,349,440,349,440,349,523,440,587,493,587,493,523,440,440,349,440,349
,440,349,493,391,523,440
230 DATA 493,391,493,391,493,391,493,391,493,391,30000,30000,391,329,493,391,587
,493,587,493,523,440,493,391
240 DATA 440,349,440,349,493,391,523,440,659,523,659,523,659,523,30000,30000,0,0
250 RANDOMIZE
260 OPTION BASE 1
270 DIM C$(4),CF(4),PT$(4),SD(4)
280 DATA 128,132,136,140,293,261,293,329
290 RESTORE 280
300 FOR I=1 TO 4 :: READ CF(I):: NEXT I :: FOR I=1 TO 4 :: READ SD(I):: NEXT I
310 CALL CLEAR :: CALL SCREEN(2):: CALL GRAFIK :: FOR I=0 TO 14 :: CALL COLOR(I,
1,1):: NEXT I
320 CALL BILD :: CALL COLOR(0,2,2,1,11,2,2,16,2,5,9,2,6,13,15,9,11,2,10,11,2,11,
11,2,12,14,8)
330 CALL COLOR(3,13,2,4,13,2)
340 CALL CHPAT(PT$())
350 LV=1 :: CALL TITEL(LV)
360 BUG=0 :: ZV=0 :: P=0 :: GS=5 :: CALL PUNKTE(P):: CALL HCHAR(23,26,121,GS)::
CALL LEVEL(LV):: RESTORE
370 ZV=ZV+1 :: IF ZV>3 THEN ZV=1
380 ON ZV GOSUB 750,820,890
390 GOSUB 1020 :: CALL COLOR(#2,8)
400 CALL MOTION(#2,LV,RND*30-15)
410 IF STE THEN 430
420 STE=1 :: CALL POSITION(#2,XA,YA):: CALL SPRITE(#4,60,14,XA+16,YA,5+LV,0)
430 FOR LOOP=1 TO 4
440 READ M1,M2 :: IF M1=0 THEN RESTORE :: GOTO 440
450 CALL SOUND(-800,M1,5,M2,5)
460 CALL C1 :: CALL PATTERN(#2,CF(LOOP)):: CALL JOYST(1,X,Y):: CALL MOTION(#1,0,
5*X):: CALL PEEK(-31877,A):: IF (A AND 32)=32 THEN 720
470 CALL KEY(1,T,S):: IF T=18 THEN SKIP=1 :: GOTO 530
480 CALL C2 :: CALL JOYST(1,X,Y):: CALL MOTION(#1,0,5*X):: CALL KEY(1,T,S):: IF
T=18 THEN SKIP=2 :: GOTO 530
490 CALL POSITION(#2,XA,YA):: IF XA>=150 THEN 740
500 CALL POSITION(#4,XA,YA):: IF XA>150 THEN CALL DELSPRITE(#4):: STE=0
510 NEXT LOOP
520 GOTO 400
530 CALL MOTION(#1,0,0):: CALL POSITION(#1,X1,Y1):: CALL SPRITE(#3,124,11,137,Y1
,-128,0)
540 CALL SOUND(-150,-5,0):: CALL POSITION(#2,X2,Y2):: IF ABS(Y1-Y2)<16-LV THEN 5
80
550 CALL DELSPRITE(#3):: READ M1,M2 :: IF M1=0 THEN RESTORE :: GOTO 550
560 CALL SOUND(-800,M1,5,M2,5)
570 ON SKIP GOTO 480,490
580 CALL DELSPRITE(#3):: CALL PATTERN(#2,36):: FOR I=0 TO 5 :: CALL SOUND(-150,-
7,I):: NEXT I
590 CALL PATTERN(#2,40):: FOR I=6 TO 10 :: CALL SOUND(-150,-7,I):: NEXT I
600 CALL PATTERN(#2,92):: FOR I=11 TO 15 :: CALL SOUND(-150,-7,I):: NEXT I :: CA
LL DELSPRITE(#2,#4)
610 P=P+10*LV :: CALL PUNKTE(P):: STE=0 :: BUG=BUG+1 :: IF BUG=6+2*LV THEN 620 E
LSE 390
620 CALL SOUND(-10,30000,0):: LV=LV+1 :: IF LV>10 THEN LV=10
630 CALL LEVEL(LV)
640 BUG=0 :: CALL COLOR(1,INT(RND*13)+3,2,2,INT(RND*13)+3,2):: GOTO 370
650 CALL MOTION(#1,0,0):: CALL PATTERN(#1,80):: FOR I=0 TO 5 :: CALL SOUND(-200,
-5,I):: NEXT I
660 CALL PATTERN(#1,84):: FOR I=6 TO 10 :: CALL SOUND(-200,-5,I):: NEXT I
670 CALL PATTERN(#1,88):: FOR I=11 TO 15 :: CALL SOUND(-150,-5,I):: NEXT I :: CA
LL DELSPRITE(#1)
680 READ M1,M2 :: IF M1=0 THEN RESTORE :: GOTO 680
690 CALL SOUND(-800,M1,5,M2,5)
700 GS=GS-1 :: CALL KANONE(GS):: IF GS=0 THEN 1040
```

```
710 CALL SPRITE(#1,76,16,153,193):: CALL DELSPRITE(#4):: IF STE<>2 THEN STE=0 ::
 GOTO 390 ELSE STE=0 :: GOTO 470
720 CALL COINC(#1,#4,16,ST):: IF ST THEN CALL DELSPRITE(#4):: CALL MOTION(#2,0,R
ND*20-10):: STE=2 :: GOTO 650
730 GOTO 740
740 CALL DELSPRITE(#2,#4):: GOTO 650
750 C$(1)="00003078FC8E0603010301020C00000000C1E3F7160C080C080403"
760 CALL SOUND(600,349,0)
770 C$(2)="0000003078FCC68703010204000000000000C1E3F63E1C080402"
780 CALL SOUND(600,261,0)
790 C$(3)="00000003078FCCE870305000000000000000000C1E3F73E1C0A"
800 CALL SOUND(600,293,0)
810 GOTO 950
820 C$(1)="00003078CD861F3261C68C0C0602000000000C1EB361F84C86633130604"
830 CALL SOUND(600,349,0)
840 C$(2)="0060F0180D063F72C30E18181000000000060F18B060FC4EC370181808"
850 CALL SOUND(600,261,0)
860 C$(3)="C06030180D067FC6830E1830000000003060C18B060FE63C170180C"
870 CALL SOUND(600,293,0)
880 GOTO 950
890 C$(1)="0000000C1E3F73E140C08000000000000000000000183C2E77E3C18"
900 CALL SOUND(600,349,0)
910 C$(2)="000000000C1E3F7321400000000000000000000000086CCFA783"
920 CALL SOUND(600,261,0)
930 C$(3)="0000000000C1E33112000000000000000000000000008CC8FC3"
940 CALL SOUND(600,293,0)
950 C$(4)=C$(2)
960 CALL SOUND(600,261,0)
970 FOR I=1 TO 4 :: CALL CHAR(4*I+124,C$(I)):: CALL SOUND(300,SD(I),0):: NEXT I
980 CALL SOUND(300,349,0)
990 CALL DELSPRITE(#1):: CALL MAGNIFY(3):: CALL SOUND(300,261,0)
1000 CALL SPRITE(#1,76,16,153,193):: CALL SOUND(300,174,0):: RESTORE
1010 RETURN
1020 CALL SPRITE(#2,132,11,1,1,0,126,#3,132,11,1,1,0,-126)
1030 FOR I=5 TO 0 STEP -1 :: CALL SOUND(-50,-6,I):: NEXT I :: CALL DELSPRITE(#3)
 :: CALL MOTION(#2,0,0):: RETURN
1040 CALL DELSPRITE(ALL):: STE=0 :: CALL CHARBACK(PT$( )):: GOTO 350
1050 SUB BILD
1060 DATA 20,10,72,1,21,10,73,1,18,16,72,1,19,16,73,3,18,24,72,1,19,24,73,3,12,2
5,72,1,13,25,73,5
1070 DATA 18,25,74,4,17,26,74,5,19,27,74,3,10,28,72,1,11,28,73,8,19,28,74,3
1080 DATA 14,29,74,8,14,30,74,8,12,31,72,1,13,31,73,1,14,31,74,8,17,32,74,5
1090 DATA A@B,CGGGD,EGGGF,17,7,15,13,9,22
1100 DATA adbef,cbbbbb9,`bbbbbh,bbbbbrs,obbbtvi,nbrsuwj,mp9lk
1110 RESTORE 1060
1120 CALL HCHAR(1,1,31,768)
1130 FOR I=1 TO 100 :: CALL VCHAR(INT(RND*20)+1,INT(RND*31)+1,INT(RND*1.5)+45)::
 NEXT I
1140 FOR I=1 TO 12 :: CALL VCHAR(INT(RND*20)+1,INT(RND*31)+1,33):: NEXT I
1150 FOR I=1 TO 20 :: READ Z,S,C,A :: CALL VCHAR(Z,S,C,A):: NEXT I
1160 READ C1$,C2$,C3$
1170 FOR I=1 TO 3 :: READ Z,S
1180 DISPLAY AT(Z,S)SIZE(3):C1$
1190 DISPLAY AT(Z+1,S-1)SIZE(5):C2$
1200 DISPLAY AT(Z+2,S-1)SIZE(5):C3$
1210 NEXT I
1220 CALL HCHAR(22,1,120,96)
1230 READ C1$ :: DISPLAY AT(4,4)SIZE(5):C1$
1240 FOR I=5 TO 9 :: READ C1$ :: DISPLAY AT(I,3)SIZE(7):C1$ :: NEXT I
1250 READ C1$ :: DISPLAY AT(10,4)SIZE(5):C1$
1260 SUBEND
1270 SUB C1 :: CALL COLOR(5,9,2):: SUBEND
1280 SUB C2 :: CALL COLOR(5,10,2):: SUBEND
1290 SUB STELL(LV):: CALL JOYST(1,X,Y):: IF X OR Y THEN LV=LV+1
1300 IF LV>10 THEN LV=1
1310 DISPLAY AT(1,1)SIZE(1):USING "#":LV-1
1320 SUBEND
1330 SUB LEVEL(LV):: CALL DELSPRITE(ALL):: CALL MAGNIFY(2):: CALL SPRITE(#1,47+L
V,2,33,49):: SUBEND
1340 SUB PUNKTE(P):: DISPLAY AT(1,22)SIZE(7):USING "#######":P :: SUBEND
1350 SUB KANONE(GS)
1360 CALL HCHAR(23,26+GS,120)
1370 SUBEND
1380 SUB CHPAT(PT$( ))
1390 FOR I=1 TO 4 :: PT$(I)="" :: FOR K=0 TO 3 :: CALL CHARPAT(124+I*4+K,A$):: P
T$(I)=PT$(I)&A$ :: NEXT K :: NEXT I :: SUBEND
1400 SUB CHARBACK(PT$( ))
```

```
1410 FOR I=1 TO 4 :: CALL CHAR(124+4*I,PT$(I)):: NEXT I :: SUBEND
1420 SUB TITEL(LV)
1430 DATA 128,8,25,129,132,8,25,161,128,5,23,131,132,5,23,163
1440 DATA 136,8,73,105,140,8,73,137,136,14,71,107,140,14,71,139
1450 RESTORE 1430
1460 CALL MAGNIFY(4):: FOR I=1 TO 8 :: READ C,F,Z,S :: CALL SPRITE(#I,C,F,Z,S)::
 NEXT I
1470 RESTORE
1480 CALL C2 :: CALL KEY(1,T,S):: IF T=18 THEN 1550
1490 READ M1,M2 :: IF M1=0 THEN RESTORE :: GOTO 1490
1500 CALL SOUND(350,M1,5,M2,5):: CALL STELL(LV)
1510 CALL C1 :: CALL KEY(1,T,S):: IF T=18 THEN 1550
1520 READ M1,M2 :: IF M1=0 THEN RESTORE :: GOTO 1490
1530 CALL SOUND(350,M1,5,M2,5):: CALL STELL(LV)
1540 GOTO 1480
1550 CALL DELSPRITE(ALL):: CALL MAGNIFY(3):: CALL SPRITE(#1,76,16,153,193)
1560 DISPLAY AT(1,1)SIZE(1):"!"
1570 SUBEND
1580 SUB GRAFIK
1590 DATA ,000000081C3E1C08,00282828,0028287C287C2828,8140201109C4320F,F10205091
020418,0182840813A4488F
1600 DATA F040201088840281,0040200109C40008,9102000900204 1,0082040003A44083,8040
0010880002 01,000000000030102,00001008
1610 DATA 00000000002,0000040810204,3C6666666666663C,183878181818183C,3C6666060C
18307E,3C66061C1C06663C,6C6C6C6C7E0C0C0C
1620 DATA 7E60607C0606663C,1C3060607C66663C,7E7E060C1830303,3C66667C3C66663C,3C6
666663E060C38,0010101038387C,000030300030102
1630 DATA 010302050502013,7878D4EC783,80C0C0E0E0C08,000C161A3B371E0C,000000007EF
F00FF,0000000000071C71,0000000000E0388E
1640 DATA 010307070F1F3C3D,80C0E0E0F0F8FCFC,7D7D7979F1E3C3FF,FEFEFEFEFFFFFFFF,FF
FFFFFFFFFFFFFF,00003F3F3F3F3F3F,3F3F3F3F3F3F3F3F
1650 DATA FFFFFFFFFFFFFFFF,0044485060504844,0101010101030306,45EDABFABAF6B7F3,80
80808080C0C06,A2B7D55F5D6FEDCF,2882008101100346
1660 DATA 050CAB22BAD0B7F3,144100818090C24,20B7D05B456DE10F,0002008801100246,010
CAA20BAD0B621,004010010090424,2097401B446DE108
1670 DATA 000000000010004,0108A2008200A401,00000000009002,2011400244010008,00400
00008000008,80020009000041,0000040002A40002
1680 DATA 00400010000002,3F3F3F7F7F7F7FFF,00000003071F3F7F,FFFFFFFFFFFFFFFF,0001
0307070F1F1F,011FFFFFFFFFFFFF,80F8FFFFFFFFFFFF
1690 DATA 000000C0E0F8FCFE,0080C0E0E0F0F8F8,FCFCFCFEFEFEFEFF,FFFEFEFEFEFCFCFC,F8
F8F0E0E0C08,FEFCF8E0C,3F0E3B073FFFF88
1700 DATA 7F3F1F0703,1F1F0F07070301,FF7F7F7F7F3F3F3F,FC70DCE0FCFF1F01,FFF0000000
00FFFF,FF8FC7F183CFF1FF,9FCFE9E3F1C7FFFF
1710 DATA FFFFFF8387F09EFC,C3F380FFFFF0FFFF,FFEFC787070F3F7,C33FE1870F1FFFFF,FF0
0FF00FF00FF,FFEFEFEFC7C783FF,0000007C0810207C
1720 DATA 0018202040202018,0101010101030501,03070D18,8080808080C0A08,C0E0B018,3C
62A12121212122,3E21202020A0613F
1730 DATA 0C1211111214181,1038A8A8A8A84582,,0044444444C54638,,00708888F080887,F0
888484848488F
1740 DATA 88848282828284F8,0000000000000042,424242424242423C,000000000000003E,40
40405E4242423C,000000000000003E,4040403C0202027C
1750 RESTORE 1590
1760 FOR I=32 TO 143 :: READ C$ :: CALL CHAR(I,C$):: NEXT I
1770 SUBEND
```

Perlentaucher

TI 99/4A

Sind Sie schon einmal Perlentaucher in der Südsee gewesen? Nein? Mit Hilfe eines TI 99/4A, EXTENDED BASIC Modul und Joysticks können Sie es werden.

Nach der Eingabe des Programms starten Sie es mit „RUN", und das Titelbild erscheint zusammen mit einer Melodie.

Ziel des Spieles ist es, einen Perlentaucher auf den Grund des Meeres zu steuern und Perlen aufsammeln lassen. Am Anfang des Spiels steht der Taucher auf einem Boot. Sie müssen nun versuchen, dieses Männchen durch einen Schwarm gefräßiger Fische sicher zu den Perlengründen zu bringen. Dabei müssen Sie darauf achten, daß Sie die Fische nicht berühren und Ihre Sauerstofftanks sich nicht leeren. Geschieht dies trotzdem, verlieren Sie eines Ihrer fünf Leben. Hat der Taucher den Meeresboden erreicht, muß eine Perle aufgenommen und zum Boot zurückgebracht werden. Nachdem die zehnte Perle zum Boot gebracht wurde, beginnt eine neue Runde, die noch schwieriger ist.
Die Steuerung des Spiels erfolgt mit dem Joystick. Der Taucher kann in acht verschiedene Richtungen gesteuert werden. Die Sauerstofftanks können nur auf dem Boot aufgefüllt werden.
Die Punkte verteilen sich wie folgt:
- große Perle 100 Punkte,
- mittlere Perle 50 Punkte,
- kleine Perle 25 Punkte.

Von den kleinen Perlen gibt es 5, von den mittleren 3 und von den großen 2.

```
100 ! ***********************
110 !*************************
120 !** (C)22.11.84  BY     **
130 !**       JUERGEN       **
140 !** WOHLFAHRT-MAHN      **
150 !**                     **
160 !*************************
170 !**                     **
180 !**       TI99/4A       **
190 !** EX.BASIC+JOYSTICK   **
200 !**                     **
210 !*************************
220 ! ***********************
```

Der Bildschirm ist folgenderweise aufgebaut:
Oben links steht die Anzeige für die Punkte, rechts daneben ist die Anzeige für den High-Score. Darunter befindet sich die Anzeige der noch vorhandenen Taucher. Daneben wird angezeigt, ob Sie eine Perle aufgenommen haben. Unter der Luftanzeige befindet sich die Rundenanzeige.

Einige Programmzeilen sind so lang, daß der Computer sie nicht annimmt. In diesem Fall müssen Sie die Zeile mit „enter" beenden und mit „FCTN+REDO" zurückholen. Sie können die Zeile nun ganz normal beenden.



```
230 !
240 !
250 GOTO 290 :: GE :: RU :: M$ :: I :: PU :: PA :: PE :: LE :: WW :: MM :: HPU :
: LW :: AA :: BB :: CC :: DD :: X :: Y :: K :: S :: C
260 X1 :: Y1 :: BO :: P1 :: P2 :: ZU :: ZZ :: LA :: TON
270 CALL CHAR :: CALL CHARPAT :: CALL COLOR :: CALL SCREEN :: CALL GCHAR :: CALL
 HCHAR :: CALL DELSPRITE :: CALL SOUND :: CALL POSITION :: CALL LOCATE :: CALL P
ATTERN :: CALL MOTION
280 CALL CLEAR :: CALL MAGNIFY :: CALL SPRITE :: CALL KEY :: CALL JOYST :: CALL
COINC
290 !@P-
300 CALL SCREEN(2):: FOR I=2 TO 8 :: CALL COLOR(I,15,2):: NEXT I :: CALL CLEAR :
: RANDOMIZE
310 CALL CHAR(64,"3C4299A1A199423C",136,"3C42B1B181B1423C")
320 DISPLAY AT(10,2):"P E R L E N T A U C H E R"
330 DISPLAY AT(12,3):" @1984 BY JUERGEN" :: DISPLAY AT(14,5):"WOHLFAHRT-MAHN"
340 FOR I=1 TO 12 :: CALL SPRITE(#I,136,INT(RND*14)+3,230,I*20,-INT(RND*10),0)::
 NEXT I
350 CALL SOUND(100,440,0):: CALL SOUND(100,550,0):: CALL SOUND(100,660,0):: CALL
 SOUND(200,790,0)
360 GOSUB 1900 :: DISPLAY AT(24,2):"TASTE DRUECKEN"
370 CALL KEY(0,K,S):: IF S=0 THEN 370
380 CALL SOUND(100,790,0):: CALL SOUND(100,660,0):: CALL SOUND(100,550,0):: CALL
 SOUND(200,440,0)
390 CALL CLEAR
400 CALL DELSPRITE(ALL)
410 DISPLAY AT(10,2):"BRAUCHEN SIE EINE ANLEITUNG              J N
        ?"
420 CALL KEY(0,K,S):: IF S=0 THEN 420 ELSE IF K=78 OR K=110 THEN 560
430 CALL CLEAR
440 DISPLAY AT(1,6):"PERLENTAUCHER" :: DISPLAY AT(3,2):"SIE MUESSEN EINEN TAUCHE
R" :: DISPLAY AT(5,2):"DURCH EINEN FISCHSCHWARM ZU"
450 DISPLAY AT(7,2):"PERLEN GRUENDEN STEUERN." :: DISPLAY AT(9,2):"DIESES IST KE
INE LEICHTE" :: DISPLAY AT(11,2):"AUFGABE, DA DIE FISCHE"
460 DISPLAY AT(13,2):"SCHNELL UND GEFRAESSIG SIND" :: DISPLAY AT(24,30):">>"
470 CALL KEY(0,K,S):: IF S=0 THEN 470
480 CALL CLEAR
490 DISPLAY AT(1,6):"PERLENTAUCHER" :: DISPLAY AT(3,2):"STEUERUNG:" :: DISPLAY A
T(5,2):"SIE MUESSEN IHREN TAUCHER,"
500 DISPLAY AT(7,2):"MIT HILFE DES JOYSTICK 1," :: DISPLAY AT(9,2):"UEBER EINE P
ERLE STEUERN" :: DISPLAY AT(11,2):"UND DANN DEN AKTIONSKNOPF"
510 DISPLAY AT(13,2):"DRUECKEN, WENN SIE EINE " :: DISPLAY AT(15,2):"PERLE AUFGE
NOMMEN HABEN," :: DISPLAY AT(17,2):"MUESSEN SIE SIE ZUM BOOT"
520 DISPLAY AT(19,2):"ZURUECK BRINGEN" :: DISPLAY AT(24,30):">>"
530 CALL KEY(0,K,S):: IF S=0 THEN 530
540 CALL CLEAR
550 CALL SOUND(100,440,0):: CALL SOUND(100,550,0):: CALL SOUND(100,660,0):: CALL
 SOUND(200,780,0)
560 !ZEICHEN DEF.
570 CALL CLEAR
580 CALL CHARPAT(70,M$):: CALL CHAR(74,M$):: CALL CHARPAT(65,M$):: CALL CHAR(87,
M$):: CALL CHARPAT(89,M$):: CALL CHAR(81,M$)
590 CALL CHAR(96,"00000302030107080B0B030202020600000080808000C0A0A0808080800C0
A0")!MANN STEIGEN
600 CALL CHAR(100,"000602020203080B0B070103020300000C08080808A0A0A0C0008080800
000")!MANN SINKEN
610 CALL CHAR(104,"0000000000011F1B0000000000000000000000000048AF0E41A000000000
000")!MANN LINKS
620 CALL CHAR(108,"000000020510F27580000000000000000000000080F8D800000000000000
000")!MANN RECHTS
630 CALL CHAR(112,"0000000000040607070604000000000000000000078FCF6FEFC780000000
000")!FISCH 1
640 CALL CHAR(116,"000000000080D0F0D08000000000000000000000F8FCF6FCF800000000
000")!FISCH 2
650 CALL CHAR(120,"00000000001E3F6F7F3F1E00000000000000000002060E0E060200000000
000")!FISCH 3
660 CALL CHAR(124,"00000000001F3F6F3F1F000000000000000000010B0F0B01000000000000
000")!FISCH 4
670 CALL CHAR(128,"3C7EFFFFFFFF7E3C00000000387C7C7C380000000000183C3C1800000000000
000")!PERLEN
680 CALL CHAR(136,"00FF7F3A1F0F0703E4FFFFFFFFFFFFFFE7FFFFFFFFFFFFFF80FCFCFCFCFCF
0F0")!BOOT
690 CALL CHAR(140,"000000000003F6400000000000FFE70000000000000000000000000000
```

```
000")!BOOT
700 CALL CHAR(32,"FFFFFFFFFFFFFFFF",40,"FFFFFFFFFFFFFFFF",88,"FFFFFFFFFFFFFFFF")
710 CALL CHAR(64,"080808080808080800000000000000000000000000000000000000000000
000")!ZEIGER
720 CALL CLEAR
730 !BILDSCHIRM-MASKE
740 CALL COLOR(2,8,8,1,5,1,8,9,1,13,11,5,14,7,8,7,2,1):: CALL SCREEN(6)
750 FOR I=3 TO 7 :: CALL COLOR(I,2,8):: NEXT I
760 RU,PU,PA,PE=0 :: LE=5 :: WW=24 :: MM=128
770 CALL DELSPRITE(ALL):: CALL CLEAR
780 CALL HCHAR(1,1,40,192):: CALL HCHAR(23,1,88,64):: CALL HCHAR(6,15,136)
790 CALL HCHAR(6,16,137):: CALL HCHAR(6,17,138):: CALL HCHAR(6,18,139)
800 CALL HCHAR(5,16,140):: CALL HCHAR(5,17,141)
810 CALL MAGNIFY(3)
820 DISPLAY AT(1,1)SIZE(11):"PUNKTE:";PU :: DISPLAY AT(1,15)SIZE(14):"HI-PUNKTE:
":HPU
830 DISPLAY AT(2,3)SIZE(5):"LUJT:"
840 CALL HCHAR(2,10,88,15)
850 CALL SPRITE(#28,64,16,9,190)
860 CALL SPRITE(#27,96,2,10,230):: DISPLAY AT(3,26)SIZE(2):LE
870 !SPIEL
880 RU=RU+1 :: GE=RU*1.5+3 :: LU=121 :: DISPLAY AT(3,2)SIZE(8):"RUNDE:";RU
890 RESTORE 900
900 DATA 112,16,1,120,6,-1,112,3,1,120,11,-1,112,4,1,120,15,-1,112,3,1
910 FOR I=2 TO 22 STEP 3 :: READ AA,BB,CC :: DD=INT(RND*3):: CALL SPRITE(#I,INT(
RND*2)*4+AA,BB,I*3.7+42,10,0,GE*CC+DD)
920 CALL SPRITE(#I+1,INT(RND*2)*4+AA,BB,I*3.7+42,96,0,GE*CC+DD):: CALL SPRITE(#I
+2,INT(RND*2)*4+AA,BB,I*3.7+42,182,0,GE*CC+DD):: NEXT I
930 CALL SPRITE(#23,124,2,130,200,0,-GE-7)
940 CALL SPRITE(#24,116,2,141,100,0,GE+10)
950 CALL SPRITE(#25,124,2,157,200,0,-GE-13)
960 GOSUB 1440
970 CALL SPRITE(#1,96,2,24,128)
980 CALL SOUND(100,220,0):: CALL SOUND(100,330,0):: CALL SOUND(100,500,0)
990 !SPIEL HAUPTPROGRAMM
1000 WW=24 :: MM=128
1010 CALL JOYST(1,X,Y):: CALL KEY(1,K,S):: LU=LU-1 :: CALL LOCATE(#28,9,LU+69)::
 IF LU<0 THEN 1610
1020 CALL COINC(ALL,C):: IF C THEN 1610
1030 IF K=18 THEN 1330
1040 IF X=-4 THEN 1090 ELSE IF X=4 THEN 1150
1050 IF Y=4 THEN 1210 ELSE IF Y=-4 THEN 1280
1060 CALL LOCATE(#1,WW,MM)
1070 CALL COINC(ALL,C):: IF C THEN 1610
1080 GOTO 1010
1090 !SCHWIMMEN LINKS
1100 CALL POSITION(#1,X1,Y1):: IF X1=48 AND Y=4 OR X1=168 AND Y=-4 OR X1<48 THEN
 1060 ELSE IF Y1=8 THEN 1060
1110 IF X1>169 AND Y=-4 THEN 1060
1120 CALL PATTERN(#1,104)
1130 MM=MM-8 :: WW=WW-2*Y :: CALL SOUND(-1,-1,0)
1140 GOTO 1060
1150 !SCHWIMMEN RECHTS
1160 CALL POSITION(#1,X1,Y1):: IF X1=48 AND Y=4 OR X1=168 AND Y=-4 OR X1<48 THEN
 1060 ELSE IF Y1=240 THEN 1060
1170 IF X1>169 AND Y=-4 THEN 1060
1180 CALL PATTERN(#1,108)
1190 WW=WW-2*Y :: MM=MM+8 :: CALL SOUND(-1,-1,0)
1200 GOTO 1060
1210 !MANN STEIGEN
1220 CALL POSITION(#1,X1,Y1)
1230 IF X1<>48 THEN 1250 ELSE CALL GCHAR(INT(X1/8),INT(Y1/8)+1,BO)
1240 IF BO<>137 AND BO<>138 THEN 1060 ELSE 1800
1250 CALL PATTERN(#1,96)
1260 IF X1<48 THEN 1270 ELSE WW=WW-8 :: CALL SOUND(-1,-1,0)
1270 GOTO 1060
1280 !MANN SINKEN
1290 CALL POSITION(#1,X1,Y1):: IF X1=168 THEN 1060 ELSE IF X1=24 THEN WW=WW+16
1300 CALL PATTERN(#1,100)
1310 WW=WW+8 :: CALL SOUND(-1,-1,0)
1320 GOTO 1060
1330 !PERLEN AUFNAHME
1340 IF PE<>0 THEN 1010
1350 CALL POSITION(#1,X1,Y1):: CALL GCHAR(INT(X1/8)+.75,INT(Y1/8)+.75,P1)
1360 IF P1=32 OR P1=88 OR P1=40 THEN CALL SOUND(-1,770,0):: GOTO 1370 ELSE PE=P1
 :: GOTO 1390
```

```
1370 CALL GCHAR(INT(X1/8)+.75,INT(Y1/8)+1.75,P2)
1380 IF P2=32 OR P2=88 OR P2=40 THEN CALL SOUND(-1,770,0):: GOTO 1060 ELSE PE=P2
 :: CALL HCHAR(INT(X1/8)+.75,INT(Y1/8)+1.75,32)
1390 CALL SOUND(100,220,0):: CALL SOUND(100,330,0):: CALL SOUND(100,440,0)
1400 CALL HCHAR(INT(X1/8)+.75,INT(Y1/8)+.75,32)
1410 CALL SOUND(100,440,0):: CALL SOUND(100,330,0):: CALL SOUND(200,220,0)
1420 CALL HCHAR(3,32,PE)
1430 GOTO 1010
1440 !
1450 FOR I=1 TO 5
1460 ZU=INT(RND*29)+3
1470 CALL GCHAR(22,ZU,ZZ):: IF ZZ<>32 THEN 1460 ELSE CALL HCHAR(22,ZU,130)
1480 NEXT I
1490 FOR I=1 TO 3
1500 ZU=INT(RND*29)+3
1510 CALL GCHAR(22,ZU,ZZ):: IF ZZ<>32 THEN 1500 ELSE CALL HCHAR(22,ZU,129)
1520 NEXT I
1530 CALL SOUND(-200,330,0,550,0,880,0)
1540 FOR I=1 TO 2
1550 ZU=INT(RND*29)+3
1560 CALL GCHAR(22,ZU,ZZ):: IF ZZ<>32 THEN 1550 ELSE CALL HCHAR(22,ZU,128)
1570 NEXT I
1580 CALL SOUND(-200,440,0,660,0,990,0)
1590 PA=10 :: RETURN
1600 !
1610 !LEBEN VERLUST
1620 CALL SOUND(100,220,0):: CALL SOUND(100,175,0):: CALL SOUND(200,-3,0)
1630 LU=121 :: LE=LE-1 :: IF PE<>0 THEN PE=0 :: PA=PA-1 :: CALL HCHAR(3,32,40)
1640 DISPLAY AT(3,26)SIZE(2):LE
1650 IF LE=0 THEN 1700 ELSE IF PA=0 THEN 880
1660 GOTO 970
1670 !
1680 !ENDE
1690 !
1700 IF PU>HPU THEN HPU=PU :: DISPLAY AT(1,25)SIZE(5):HPU
1710 DISPLAY AT(10,4):"PLWQ WGWIN  <Q:N>? "
1720 DISPLAY AT(12,9)BEEP:"GWME OVER"
1730 CALL KEY(0,K,S):: IF S=0 THEN 1750 ELSE IF K=74 OR K=106 OR K=89 OR K=121 T
HEN 740
1740 STOP
1750 FOR I=1 TO 100 :: NEXT I
1760 DISPLAY AT(12,9):"          "
1770 CALL KEY(0,K,S):: IF S=0 THEN 1790 ELSE IF K=74 OR K=106 OR K=89 OR K=121 T
HEN 740
1780 STOP
1790 FOR I=1 TO 100 :: NEXT I :: GOTO 1720
1800 !AUF DEM BOOT
1810 LU=121 :: CALL PATTERN(#1,96)
1820 CALL SOUND(-200,440,0,220,0):: WW=WW-24 :: IF PE=0 THEN 1060
1830 CALL LOCATE(#1,WW,MM):: CALL SOUND(100,330,0,440,0)
1840 IF PE=128 THEN PU=PU+100*RU ELSE IF PE=129 THEN PU=PU+50*RU ELSE IF PE=130
THEN PU=PU+25*RU
1850 CALL SOUND(100,392,0):: CALL SOUND(100,392,0,330,0):: CALL SOUND(100,392,0,
330,0):: CALL SOUND(200,440,0,392,0)
1860 CALL SOUND(200,392,0,330,0)
1870 PE=0 :: PA=PA-1 :: IF PA=0 THEN CALL DELSPRITE(ALL):: PU=PU+150*RU :: DISPL
AY AT(1,8)SIZE(5):PU :: GOTO 880
1880 DISPLAY AT(1,8)SIZE(5):PU :: CALL HCHAR(3,32,40)
1890 GOTO 1060
1900 !MUSIK
1910 RESTORE 1970
1920 READ LA,TON :: IF LA=0 THEN 1950
1930 CALL SOUND(LA,TON,5,TON+5,10)
1940 GOTO 1920
1950 RETURN
1960 !DATA MUSIK
1970 DATA 200,659,200,659,200,659,600,523,200,587,200,587,200,587,600,493
1980 DATA 200,659,200,659,200,659,200,659,200,698,200,698,200,698,200,659
1990 DATA 200,1046,200,1046,200,1046,600,880,200,659,200,659,200,659
2000 DATA 200,659,200,698,200,698,200,698,200,659,200,987,200,987,200,987
2010 DATA 600,830,200,1318,200,1318,200,1174,600,1046,200,1318,200,1318,200,1174
,600,1046
2020 DATA 200,1318,200,1318,200,1174,600,1046,600,880,600,1318
2030 DATA 0,0
2040 REM DAS PROGRAMM IST    9468 BYTES LANG
```

Caverns of Death

Commodore 64 (mit Joystick/Port 2)

Programm des Monats

CAVERNS OF DEATH enthält mehrere Maschinenspracheroutinen, die einen fließenden Bewegungsablauf ermöglichen und besonders bei höheren Schwierigkeitsgraden temporeiches Spiel garantieren.

Der scheinbar unbewohnte und lebensfeindliche Planet Tresed am Rande des Andromeda-Nebels stellt eine nahezu perfekte Tarnung des Kampfsterns Promes dar ...

Als Gefangener der Tresaner nehmen Sie die erste Gelegenheit einer Flucht war und versuchen mit einem Raumer durch die vordersten Anlagen zu entkommen: ... 3 todbringende Höhlen mit raffinierten Abwehrsystemen ...

Kollidieren Sie dabei mit feindlichen Objekten, der Höhlenwand oder überschreiten das in Bild 1 gesetzte Limit von 99 Zeiteinheiten, so wird ihr Raumer, der je Spiel nur 9mal ersetzt wird, zerstört. Am Ende jeder Höhle befindet sich ein Durchgang, in dem Sie so weit wie möglich nach oben fliegen müssen, um das nächste Bild zu erreichen (manchmal dauert es etwas länger!) ...

Bild 2 beinhaltet unter anderem eine permanente Sperre, die sich nur öffnet, wenn Sie ein ziemlich schußfestes Geschütz Ihres Gegners vernichten, das nur mit mehreren Treffern zerstört werden kann. Sollten Sie alle 3 Höhlen erfolgreich gemeistert haben, so beginnt das Spiel mit erhöhtem Schwierigkeitsgrad von neuem ...

```
0 REM ***********************************
1 REM *                                 *
2 REM *         CAVERNS OF DEATH        *
3 REM *             (C) 1984            *
4 REM *            WRITTEN BY           *
5 REM *                                 *
6 REM *         CHRISTIAN WURZER        *
7 REM *                                 *
8 REM ***********************************
9 GOTO1000
10 REM ** KOLLISION ?
11 :
15 FORT=0TO2
16 IFPEEK(53246)=1THEN25
17 IF(PEEK(V+30)AND1)=1THEN25
18 NEXT:RETURN
25 REM ** KOLLISION
26 :
27 POKEV+30,0:POKEV+31,0
29 L=L-1:POKE53246,1:POKEV,57:POKEV+1,199
30 IFL=-1THEN40
31 POKE214,22:POKE211,35:SYS58640
32 PRINT"██"STR$(L)
33 FORT=30TO1STEP-1:POKE53206,T:SYS53200:NEXT
34 POKE53246,0:RETURN
40 REM ** GAME OVER
41 :
42 POKEV+21,0:PRINT"J":PRINTTAB(15)"██GAME OVER"
44 PRINT"████    FEUERKNOPF DRUECKEN UM DAS SPIEL"
45 PRINTTAB(11)"█ERNEUT ZU STARTEN":PRINTTAB(14)"█████SCORE ="SC
46 POKE53221,33:POKE53206,INT(RND(0)*255):SYS53200
48 IFPEEK(56320)<>111THEN46
49 POKE53221,129:POKE53206,130:GOTO15000
50 REM ** BILD 1
51 :
52 GOTO30000
80 TI$="000000":HE=99:POKEV+21,15
81 GOSUB720:L=L+1:GOSUB25:GT=0
100 SYS49152
112 POKEV+30,0:POKEV+31,0
120 REM ** BILD 1 BEENDET ?
125 IFPEEK(V+1)<45THEN603
200 :
205 IFVAL(TI$)<(1+LE)THEN300
```

```
206 :
207 REM ** UHR
208 HE=HE-1:IFHE=-1THENHE=99:GOSUB25
209 POKE214,4:POKE211,24:SYS58640
210 POKE1209,160:POKE1210,160:PRINT"▓▓"HE
220 POKE1648,101:POKE1688,84:POKE1728,71:POKE1768,66:POKE53206,120:SYS53200
240 FORT=1TO5:POKE56944,T:POKE56945,T:POKE56946,T:POKE56947,T:NEXT
250 POKE1648,32:POKE1688,32:POKE1728,32:POKE1768,32:TI$="000000"
300 :
400 REM ** KOLLISION ?
401 :
405 GOSUB10:GOTO100
603 REM ** BILD 2
605 :
630 POKEV+21,0:GOTO32000
631 POKEV,56:POKEV+1,199:SC=SC+(HE*10)+100
632 POKEV+8,200:POKEV+9,173:POKE715,0
633 GOSUB720:POKEV+21,63:L=L+1:GOSUB25
640 REM ** BILD 2 BEENDET ?
641 :
642 IFPEEK(V+1)<45THEN800
643 :
645 REM ** SPERREN ?
646 IFVAL(TI$)<5THEN652
647 POKE1546,32:POKE1547,32:POKE1548,32:POKE1256,32:POKE1257,32:POKE1258,32
648 FORT=0TO(300+LE*70):NEXT
649 POKE1546,28:POKE1547,28:POKE1548,28:POKE1256,28:POKE1257,28:POKE1258,28
650 TI$="000000"
652 REM ** KOLLISION ?
653 :
654 GOSUB10
660 REM ** RAKETE DES GEGNERS
661 :
670 IFPEEK(715)<>0THEN680
675 POKEV+10,PEEK(V+8):POKEV+11,PEEK(V+9)-5
676 POKE715,255:POKE714,0
677 POKE53206,30:SYS53200
680 REM ** GEGNER VERNICHTET ?
681 :
683 FORT=0TO2:IF(PEEK(V+30)AND16)=16THEN686
684 NEXT:GOTO640
686 REM ** GEGNER VERNICHTET
687 :
688 SC=SC+100:GT=GT+1:GOSUB720
689 IFGT=4THENPOKE1136,32:POKE1137,32:POKE1138,32:POKEV+21,15:POKEV+10,0
703 GOTO640
720 REM ** SCORE
721 :
724 POKE214,22:POKE211,10:SYS58640:PRINT"▓▓"SC:RETURN
764 :
800 REM ** BILD 3
801 :
822 POKEV+21,0:GOTO33000
823 TI$="000000"
824 SC=SC+100:GOSUB720
826 POKEV+21,15:L=L+1:GOSUB25
850 REM ** SPERREN / RAKETEN ?
851 :
852 IFVAL(TI$)<5THEN900
860 POKE1267,32:POKE1268,32:POKE1269,32
862 POKE1427,32:POKE1428,32:POKE1429,32
863 POKE1642,32:POKE1682,32:POKE1722,32
864 POKE1647,32:POKE1687,32:POKE1727,32
865 POKE1651,32:POKE1691,32:POKE1731,32
870 FORT=0TO(400+L*70):NEXT
875 POKE1267,28:POKE1268,28:POKE1269,28
876 POKE1427,28:POKE1428,28:POKE1429,28
877 POKE1642,28:POKE1682,28:POKE1722,28
878 POKE1647,28:POKE1687,28:POKE1727,28
879 POKE1651,28:POKE1691,28:POKE1731,28
880 REM RAKETEN
882 FORT=1328TO1168STEP-40:POKE54272+T,1:POKET,37:SYS53200:QQ=SIN(T)
883 POKET,32:NEXT
884 FORT=1261TO1244STEP-1:POKE54272+T,1:POKET,38:SYS53200:QQ=SIN(T)
885 POKET,32:NEXT
890 TI$="000000"
```

```
900 REM ** BILD 3 BEENDET ?
901 :
902 IFPEEK(V+1)<45THEN950
910 REM ** KOLLISION ?
911 :
912 GOSUB10
913 GOTO850
950 REM ** BILD 3 BEENDET
951 :
952 :
960 SC=SC+100:POKEV+21,0:IFLE<>0THENLE=LE-1
965 PRINT"£££££↑":PRINT"£↑"
967 PRINT"£  *":PRINT"£  * []"
969 PRINT"£][£££££]"
970 PRINT"...ES FOLGT DER NAECHSTE DURCHGANG...":FORT=0TO300:NEXT
972 FORT=130TO1STEP-1:POKE53206,T:SYS53200:NEXT
973 FORT=1TO130:POKE53206,T:SYS53200:NEXT:FORT=0TO300:NEXT
976 GOTO50
1000 REM ** BITTE WARTEN
1010 :
1011 PRINT"":POKE53280,0:POKE53281,0
1012 PRINT"    [illegible]"
1014 PRINT"    [illegible]"
1016 PRINT"    [illegible]"
1017 PRINT"    [illegible]"
1018 PRINT"    [illegible]"
1022 PRINTTAB(16)"OF DEATH"
1024 PRINT"EIN ARCADE-SPIEL FUER DEN COMMODORE 64"
1025 PRINTTAB(8)"MIT JOYSTICK  (PORT 2)"
1030 PRINTTAB(13)"BITTE WARTEN"
1040 PRINTTAB(7)"DATEN WERDEN EINGELESEN"
1050 :
2000 REM ** MASCHINENPROGRAMM
2002 :
2003 DATA120,169,192,141,21,3,169,13,141,20,3,88,96
2005 DATA174,0,220,224,112,176,73,174,195,2,224,0,208,66,173,16,208,41,253,141
2006 DATA16,208,173,16,208,41,1,170,224,1,208,6,238,16,208,238,16,208,174,0,208
2007 DATA142,2,208,174,1,208,142,3,208,162,1,142,195,2,162,0,142,4,212
2008 DATA162,4,142,1,212,162,10,142,5,212,162,15,142,24,212,162,129,142,4,212
2010 DATA174,31,208,142,255,207
2011 DATA173,255,207,41,2,170,224,2,208,8,162,0,142,195,2,142,3,208
2013 DATA173,255,207,41,32,170,224,32,208,8,162,0,142,203,2,142,11,208
2015 DATA173,255,207,41,1,170,224,1,208,23,162,57,142,0,208,162,199,142,1,208
2016 DATA173,16,208,41,254,141,16,208,162,1,142,254,207,169,0,141,192,2,141,193
2103 DATA2,173,0,220,41,239,141,253,207,174,254,207,224,1,240,112
2105 DATA174,253,207,224,107,208,5,169,255,141,192,2,174,253,207,224,103
2106 DATA208,5,169,1,141,192,2
2107 DATA174,253,207,224,110,208,5,169,255,141,193,2,174,253,207,224,109,208,5
2108 DATA169,1,141,193,2,174,253,207,224,106,208,8,169,255,141,193,2,141,192,2
2109 DATA174,253,207,224,102,208,10,169,255,141,193,2,169,1,141,192,2
2110 DATA174,253,207,224,105,208,10,169,1,141,193,2,169,255,141,192,2
2111 DATA174,253,207,224,101,208,8,169,1,141,193,2,141,192,2
2119 DATA173,16,208,41,1,170,224,1,240,12,174,0,208,224,30,176,17,169,1
2120 DATA141,192,2,174,0,208,224,60,144,5,169,255,141,192,2
2150 DATA174,1,208,224,40,176,5,169,1,141,193,2,174,1,208,224,200,144,5,169,255
2151 DATA141,193,2
2182 DATA174,5,208,224,90,176,5,162,1,142,197,2,174,5,208,224,175,144,5,162,255
2183 DATA142,197,2
2184 DATA174,7,208,224,90,176,5,162,1,142,199,2,174,7,208,224,175,144,5,162,255
2185 DATA142,199,2
2193 DATA174,8,208,224,100,176,5,162,1,142,200,2,174,8,208,224,240,144,5,162
2219 DATA255,142,200,2,162,15,169,128,141,61,3,141
2220 DATA60,3,222,208,2,208,44,189,176,2,157,208,2,189,192,2,240,33,16,12,189,0
2230 DATA208,8,222,0,208,40,240,7,208,19,254,0,208,208,14,173,60,3,208,9,173,61
2240 DATA3,77,16,208,141,16,208,173,60,3,208,3,78,61,3,73,128,141,60,3,202,16
2250 DATA191,76,49,234,-1
3010 DATA162,0,142,11,212
3020 DATA162,255,142,8,212,162,10,142,12,212,162,15,142,24,212,162,129,142,11
3025 DATA212,96,-1
4080 DATA120,169,49,133,1,169,0,133,98,133,100,169,208,133,99,169,56,133,101
4082 DATA162,16,160,0,177,98,145,100,200,208,249,230,99,230,101,202,208,242
4085 DATA169,55,133,1,88,96,-1
4095 DATA1,1,3,7,15,31,127,255,255,255,255,255,255,255,255,255
4096 DATA128,128,192,224,240,248,254,255,255,252,252,240,224,192,128,128
4097 DATA255,63,63,7,3,3,1,1,,,,,,,,,8,76,247,3,3,247,76,8,0,0,0,0,0,0,0,0
4098 DATA16,16,16,56,56,56,108,68,124,108,238,198,214,84,56,16
```

```
4099 DATA16,16,16,56,56,124,56,16,0,3,15,252,252,15,3,-1
5000 REM ** EINLESEN DER DATEN
5001 :
5010 FORT=0TO1999:READQ:IFQ=-1THEN5026
5012 POKE49152+T,Q:NEXT
5026 FORT=0TO999:READQ:IFQ=-1THEN5030
5027 POKE53200+T,Q:NEXT
5030 FORT=0TO999:READQ:IFQ=-1THEN5035
5032 POKE53000+T,Q:NEXT
5035 SYS53000:POKE53272,(PEEK(53272)AND240)OR14
5036 FORT=0TO999:READQ:IFQ=-1THEN5050
5037 POKE14336+27*8+T,Q:NEXT
5050 :
6000 REM ** SPRITES
6002 :
6006 REM ** EIGENER RAUMER
6010 DATA,,,,,,,,,,60,,,102,,,195,,15,255,240,31,255,248,48,60,12,48,24,12,24
6020 DATA,24,24,,24,8,,16,,,,,,,,,,,,,,,,,,,,,,,,
6030 :
6032 REM ** BOMBE
6035 DATA,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,24,,,24,,,24,,,,,,,,,,,,,,,,,,,,,,,,
6040 REM ** GESCHUETZ (GEGNER)
6045 DATA,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,48,,,48,,8,48,64,8,48,64,56,120
6046 DATA112,127,255,248,63,255,240,31,255,224,15,3,192,
10000 REM ** EINLESEN DER SPRITES
10001 :
10010 FORT=0TO63:READQ:POKE832+T,Q:NEXT
10020 FORT=0TO63:READQ:POKE896+T,Q:NEXT
10030 FORT=0TO63:READQ:POKE960+T,Q:NEXT
15000 REM ** LEVEL
15001 :
15003 FORT=0TO100:POKE53206,T:SYS53200:NEXT
15010 PRINT"]]SIND SIE 1 = ANFAENGER,     "
15020 PRINTTAB(7)"▯  2 = FORTGESCHRITTENER ODER"
15025 PRINTTAB(7)"▯  3 = EXPERTE"
16000 GETA$
16010 IFA$="1"THENLE=4:GOTO20000
16011 IFA$="2"THENLE=2:GOTO20000
16012 IFA$="3"THENLE=0:GOTO20000
16013 GOTO16000
20000 REM ** VARIABLEN, DATEN
20001 :
20002 V=53248:L=9:SC=0:POKE53247,0:POKE53246,0:POKE56325,20
20022 POKEV+27,44:POKE706,0:POKE707,0
20023 POKE708,0:POKE709,1
20024 POKE710,0:POKE711,255
20025 POKE712,1:POKE713,0
20042 POKE688,3:POKE689,3
20043 POKE690,3:POKE691,2
20044 POKE692,3+LE:POKE693,3+LE
20045 POKE694,3+LE:POKE695,3+LE
20046 POKE696,2+LE:POKE699,2+LE
20060 POKE2040,13:POKE2041,14:POKE2042,14:POKE2043,14:POKE2044,15:POKE2045,14
20142 POKEV+39,1:POKEV+40,1:POKEV+41,1
20143 POKEV+42,1:POKEV+43,1:POKEV+44,1
20170 POKEV+16,0:POKEV,56:POKEV+1,200
20172 POKEV+3,0:POKEV+4,95:POKEV+5,110
20173 POKEV+6,255:POKEV+7,110:GOTO50
30000 REM ** BILD 1
30010 :
30090 PRINT"]I█££££££££££££££££££££   ££££££££££££££££££"
30100 PRINT"]££££££££££££££££££££   ££££££££££££££££££"
30105 PRINT"]£££££↑        ←££££   ££█TIME█£££££££££££££"
30110 PRINT"]££££↑         ←£££   ££££££££££££↑    ←££"
30115 PRINT"]£££↑           ←£↑   ←££££££££££↑      ←£"
30120 PRINT"]££↑     [££]     ←]    ££££££££↑          £"
30125 PRINT"]£↑    [££↑ ←]     ←]   ←£££££££↑          £"
30130 PRINT"]£    [£££   £]     £    £££££↑     []     £"
30135 PRINT"]£    £↑ ←   £↑    [£    ←£↑       £↑     £"
30140 PRINT"]£           £    [££             [£    [£"
30150 PRINT"]£           £    ←£↑            [££↑    ££"
30155 PRINT"]£]          £     ←]          [££↑      ££"
30160 PRINT"]£££££] [    £]     ←]  [££££££↑        [££"
30165 PRINT"]££↑  ←££    ££]    ←££££££↑            £££"
30170 PRINT"]£↑          ←↑←]    ←↑←£↑         [↑    ←£"
30175 PRINT"]£]             £                [£      £"
```

```
30180 PRINT"]££]            £]              [££]     £"
30185 PRINT"]£££]   [£]   [££]           [£££££]   [£"
30186 PRINT"]££££   ££££££££££££]        [£££££££££££££"
30187 PRINT"]££££   ££££££££££££££££££££££££££££££££££££"
30188 PRINT"]££££     ££££££££££££££££££££££££££££££££££"
30189 PRINT"]££££££££££££££££££££££££££££££££££££££££££"
30190 PRINT"]] SCORE                      RAUMER       "
30191 PRINT"]£££££££££££££££££££££££££££££££££££££££££££":POKE1983,160:POKE56255,9
31000 GOTO80
32000 REM ** BILD 2
32001 :
32017 PRINT"]]]££££££££££££££££££££££££££££££££   £££££"
32018 PRINT"]]££££££££££££££££££££££££££££££££   £££££"
32020 PRINT"]↑  ←↑   ←£££↑←↑←↑←↑←↑←££↑←££££     ]£££££"
32025 PRINT"]          £££              ££  ←££   ↑[£££"
32030 PRINT"]          £££             ←↑   ←£   [££££"
32032 PRINT"]          ←££]                ←   £££££"
32033 PRINT"]            ←£]                  ←££££"
32035 PRINT"]              £↑                    £££"
32036 PRINT"]              £]                   [£££"
32037 PRINT"]              ££                   ←£££"
32038 PRINT"]              £↑                    £££"
32039 PRINT"]              £]                   [£££"
32040 PRINT"]£]            ←↑                   ←£££"
32042 PRINT"]££    ££]                           £££"
32043 PRINT"]↑     ←£]                          [£££"
32044 PRINT"]]     [££                          ££££"
32045 PRINT"]££]   £££   ]                     [££££"
32046 PRINT"]£££   £££] [£] []              [££££££"
32047 PRINT"]££↑   ←£££££££££££££££££££££££££££££££££"
32048 PRINT"]££]    ££££££££££££££££££££££££££££££££"
32049 PRINT"]£££     £££££££££££££££££££££££££££££££££"
32050 PRINT"]£££££££££££££££££££££££££££££££££££££££££"
32051 PRINT"]] SCORE                      RAUMER       "
32052 PRINT"]£££££££££££££££££££££££££££££££££££££££££££":POKE1983,160
32053 POKE56255,9:GOTO631
33000 REM ** BILD 3
33009 :
33010 PRINT"]]]£££££££££££££     ££££££££££££££££££££££££"
33012 PRINT"]£££££££££££££     ££££££££££££££££££££££££"
33014 PRINT"]£££↑       ←££     ←£££↑ ←££↑            ←£"
33016 PRINT"]££↑         ←£      ←£↑                   £"
33018 PRINT"]££           ←]      £                    £"
33019 PRINT"]££]   []      £      £                  ]|]£"
33020 PRINT"]£££   ££     [£     [↑                    £"
33022 PRINT"]£££   £↑     £↑                           £"
33023 PRINT"]££↑   £]    [↑              ]]]          [£"
33024 PRINT"]££]   £↑    ←]    []        ]]]   [   ]   [£££"
33026 PRINT"]£££   £]      ←££££££]    [£££££  £     ←£££"
33027 PRINT"]££↑   ££       ←££££££][£££££↑  £]     ←££"
33028 PRINT"]££   ←£]       ←£££££££££££££]  ←↑      ←£"
33030 PRINT"]££]    ££]         ←£££££££££££££  []     £"
33032 PRINT"]£££]   ←££          ←£££££££££££↑  ←£]     £"
33034 PRINT"]££££     ££                         ←     £"
33035 PRINT"]££↑      ££]                            [£"
33036 PRINT"]££]    [£££]                            ←£"
33037 PRINT"]£££    £££££    [£££££££££££££]           £"
33038 PRINT"]£££    £££££] [££££££££££££££££]          [£"
33039 PRINT"]£££     ££££££££££££££££££££££££££££££££££££"
33040 PRINT"]£££££££££££££££££££££££££££££££££££££££££££"
33100 PRINT"]] SCORE                      RAUMER       "
33101 PRINT"]£££££££££££££££££££££££££££££££££££££££££££":POKE1983,28
33102 POKE56255,9:GOTO823
```

Bei Zusendung von Manuskripten und Software erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger.
Rücksendung erfolgt nur gegen Erstattung der Unkosten. Zusendungen von Software zur Veröffentlichung sollten folgendes enthalten:
Kopierfähige Kassette oder Diskette mit dem Programm (Computer-Bezeichnung), von Drucker erstelltes Listing oder Serie von Bildschirmfotos (keine Schreibmaschinenlistings) evtl. Bildschirmfotos von einem Probelauf und ausführliche Programmbeschreibung (Erklärung der Variablenliste, Beschreibung des Bildschirmaufbaus, Farbe, Grafik usw.). Für eingesandte Programmunterlagen kann keinerlei Haftung übernommen werden. IVW geprüft.

Impressum

Homecomputer
erscheint monatlich im
Tronic-Verlag, 3444 Wehretal 1

Redaktion:
Axel Gredé (verantwortlich),
Rainer Beck, Frank Brall

Freie Mitarbeiter:
Volker Becker, Holger Gredé

Gesamtherstellung:
D+V Druckhaus Dierichs Kassel

Vertrieb:
Inland (Groß-, Einzel- und Bahnhofsbuchhandel) sowie Österreich und Schweiz:
Verlagsunion
Friedrich-Bergius-Straße 20
6200 Wiesbaden
Telefon 0 61 21 / 26 60

Anfragen nicht an Vertrieb oder Druckerei, sondern nur an den Verlag!

Anzeigenleitung:
Anneke Kratzenberg

Erscheinungsweise:
Erstverkaufstag von HOMECOMPUTER ist Anfang des Monats.

Anzeigenpreise:
Bitte Mediaunterlagen anfordern.

Anzeigenannahmeschluß:
Jeweils 3 Wochen vor Erscheinungstermin.

Bezugspreise:
Einzelheft 5,50 DM
Abonnement: Inland 55,– DM im Jahr (12 Ausgaben)
Ausland: Europa 80,– DM, USA 110,– DM

Autoren, Manuskripte:
Der Verlag nimmt Manuskripte und Software zur Veröffentlichung gerne entgegen.
Sollte keine andere Vereinbarung getroffen sein, so gehen wir davon aus, daß Sie mit einem Honorar von 120,– DM pro abgedruckter Seite im Heft einverstanden sind.

War Lords
Commodore 64

Das Spiel ‚War Lords' stellt den Kampf zweier verfeindeter Burgherren dar,

die sich gegenseitig mittels zweier Katapulte einen erbitterten Kampf liefern, denn es gewinnt nur der, der zuerst die Burg des anderen vollständig zerstört hat. Um bis zur Burg des Gegners vorzudringen ist es erst einmal notwendig, sich durch den Wald des Gegners zu kämpfen, in dem seine Burg steht. Damit man dem Gegner nicht hilflos ausgeliefert ist, kann man ein Schutzschild vor dem eigenen Territorium so bewegen, daß die Geschosse des Gegners daran abprallen, und ihn nun selbst bedrohen.

Punkte gibt es natürlich auch, und zwar für die Bäume, die man dem Gegner abholzt. (Keine Sorge, denn die sind schon so vom sauren Regen geschädigt, daß man sie sowieso bald fällen müßte.) Natürlich werden die Schutzschilde mittels Joystick (1+2) bewegt, da man davon ausgehen kann, daß wohl jeder zwei Joysticks besitzt. Spieler 1 muß dabei den Joystick an Port 1 benutzen, und hat seine Burg auf der linken Seite zu beschützen, für Spieler 2 gilt es natürlich genau anders herum.

Da es mitunter sehr lange dauern kann, bis man die Burg des Gegners zerstört hat (besonders, wenn der Gegner schon etwas Übung hat), kann man auch die Möglichkeit wählen, eine bestimmte Zeit lang zu spielen, gewonnen hat dann, wer dem anderen die meisten Bäume entwurzelt hat.

Natürlich werden die Punkte der beiden Spieler fortlaufend addiert, so daß man z. B. nach einem Turnier von fünf Spielen den besten Burgherrn aller Zeiten bestimmen kann.

Startet man das Programm, so wird zuerst anhand des Inhalts der Speicherstelle 32768 festgestellt, ob die Datas schon an ihren Platz gepoket worden sind. Ist das der Fall, so springt das Programm in Zeile 1370 gleich weiter zum Titelbild. Ansonsten werden die Datas erst einmal in den Zeilen 2540–2800 an ihren Platz gepoket. Vorher wird jedoch in Zeile 1310 der Bildschirm ausgeschaltet, was während des Einlesens der Datas eine Zeitersparnis von ca. 6 % bringt. Anschließend wird der Bildschirm in Zeile 2890 wieder aktiviert und das Titelbild wird angezeigt. Durch drücken einer Taste (oder



Feuerknopf Port 1) geht es dann mit dem Spiel los. Die Abfrage dazu findet in Zeile 2970 statt.
Als nächstes werden die beiden Spieler gefragt, ob sie nach Zeit spielen wollen, d. h., daß nach einer bestimmten Zeit das Spiel automatisch aufhört, oder ob sie bis zum (bitteren Ende) durchhalten wollen. Ist ersteres gewählt, so werden die beiden Spieler außerdem noch gefragt, wie lange sie spielen wollen. Die Abfrage dazu erfolgt in den Zeilen 1410–1450.
Nun geht's aber los mit dem Spiel: Nachdem in Zeile 1480 der Bildschirm auf den Sonderzeichenmodus ab Speicherzelle 12288 umgeschaltet wird, wird in den Zeilen 1510–1540 die Umrandung hingepoket. Es folgt in den Zeilen 1580–1600 das ‚Pflanzen' des Waldes. In den Zeilen 1640–1670 werden die Burgen ‚erbaut', und in Zeile 1680 werden die Schutzschilde gesetzt. Nun wird in den Zeilen 1730–1750 der Ball ins Ziel gebracht, wobei die Richtung, in die der Ball losfliegt, zufällig bestimmt wird.
Es folgt jetzt der Programmteil, der die Hauptarbeit leistet, d. h., in den Zeilen 1790–1830 werden durch ein Maschinenprogramm die Schutzschilde der beiden Spieler bewegt, die Kugelposition neu berechnet, abgefragt, ob ein Baum oder der Rand getroffen wurde, und eventuell darauf reagiert. Die Abfrage, was genau getroffen wurde, erfolgt in den Zeilen 1870–1960. Je Art des Hindernisses wird dann eine der vier Sound-Routinen in den Zeilen 2250–2500 aufgerufen. Wird eine der Burgen getroffen, die Abfrage hierzu findet in den Zeilen 1930–1990 statt, so wird danach in Zeile 1980 abgefragt, ob die getroffene Burg vollständig zerstört wurde und das Spiel damit beendet ist, oder ob einfach nur der Ball neu ins Spiel gebracht werden muß, Zeile 1990.

Ist das Spiel schließlich beendet, entweder, weil die festgelegte Zeit überschritten oder weil eine der Burgen zerstört wurde, so springt das Programm in die Zeilen 2030–2210, wo natürlich zuerst einmal angezeigt wird, daß das Spiel beendet ist. Anschließend wird angezeigt, wer diese Runde gewonnen hat, Zeile 2090–2150. In den Zeilen 2160–2180 wird der Gesamtpunktestand ermittelt und angezeigt. Für ein neues Spiel wird in Zeile 2200 abgewartet, daß eine Taste bzw. der Feuerknopf an Port 1 gedrückt wird, um ein neues Spiel zu starten.

```
1010 rem"| Steuerzeichen
1030 rem"
1040 rem"   ■ = Crsr rechts
1050 rem"   ■ = Crsr links
1060 rem"   ■ = Crsr hoch
1070 rem"   ■ = Crsr runter
1080 rem"   ■ = Gelb
1090 rem"   ■ = Dunkelgrau
1100 rem"   ■ = Mittelgrau
1110 rem"   ■ = Hellgrau
1120 rem"   ■ = Braun
1130 rem"   ■ = Weiss
1140 rem"   ■ = Hellblau
1150 rem"   ■ = Gruen
1160 rem"   ■ = home
1170 rem"   ■ = clr home
1180 rem"   ■ = rvs on
1190 rem"   ■ = rvs off
1200 rem"
1220 rem"|                              |
1230 rem"|  Warlords Nov./Dez. 1984     |
1240 rem"|                              |
1250 rem"|  (c) and Idea: P.Paulsen     |
1260 rem"|                              |
1270 rem"|  written by: E.Schiffler     |
1280 rem"|                              |
1300 rem"
1310 poke53272,20:poke53280,0:poke53281,0:poke657,128:poke56,48:poke53265,11
1320 print"■■■■■■■"
1330 print"■■retten sie ihren koenig! der thron ist  ■in gefahr."
1340 print"■ein verfeindeter burgherr blaest zum    ■angriff auf das schloss.
1350 print"■wehren sie die schnellen felsbrocken"
1360 print"■ab, versuchen sie, die feindliche burg  ■zu zerstoeren.■"
1370 ifpeek(32768)<>169thengosub2540
1380 gosub2810
1390 print"■■player 1 = port 1
1400 print"■player 2 = port 2
1410 print"■■mach zeit (j/n)?";:poke204,0
1420 geta$:ifa$="n"thenpoke204,1:goto1470
1430 ifa$<>"j"then1420
1440 poke204,1:printa$
1450 input"■nach wievielen sekunden";se:se=se*60
1460 gosub1480
1470 se=9999999
1480 poke53272,28
1490 poke53281,0
1500 print"■■"
1510 fori=0to39:poke1024+i,6:poke1984+i,6:next
1520 fori=1to23:poke1024+i*40,7:poke1024+39+i*40,7:next
1530 poke1024,8:poke1024+39,8
1540 poke1984,8:poke1984+39,8
1550 rem"┌──────────────────────────┐
1560 rem"| Wald Pflanzen            |
1570 rem"└──────────────────────────┘
1580 fori=1to23
1590 print"■■dddddddd"spc(20)"dddddddd"
1600 next
1610 rem"┌──────────────────────────┐
1620 rem"| Burgen bauen             |
1630 rem"└──────────────────────────┘
1640 print"■■■■■■■■■■■■@b"spc(34)"@b"
1650 print"■ac"spc(34)"ac"
1660 print"■■■■■■■■■@b"spc(34)"@b"
1670 print"■ac"spc(34)"ac"
1680 sys32768
1690 ti$="000000":si=54272
1700 rem"┌──────────────────────────┐
1710 rem"| Ball ins Spiel           |
1720 rem"└──────────────────────────┘
1730 bp=1024+60:pokebp,9
1740 rx=-1:ifrnd(1)<.5thenrx=1
1750 ry=1
1760 rem"┌──────────────────────────┐
1770 rem"| Ball bewegen             |
1780 rem"└──────────────────────────┘
```

```
1790 sys32830:pokebp,32:bp=bp+rx+ry*40:p=peek(bp)
1800 q=bp-1024:s=abs(q-int(q/40)*40-20):ifp<9then1870
1810 ifti>sethen2030
1820 p=peek(bp):ifp<4then1920
1830 pokebp+54272,1:pokebp,9:goto1790
1840 rem" ┌──────────────────────────┐
1850 rem" | Hindernisse              |
1860 rem" └──────────────────────────┘
1870 ifp=4thengosub2250:pokebp,32:rx=-rx:goto1820
1880 ifs<16andrnd(1)<.1thenbp=bp-rx
1890 ifp=7thengosub2350:bp=bp-2*rx:rx=-rx:goto1820
1900 ifp=6thengosub2420:bp=bp-80*ry:ry=-ry:goto1820
1910 ifp=8thenbp=bp-80*ry-2*rx:rx=-rx:ry=-ry:goto1820
1920 pokebp,32
1930 ifp=0then1980
1940 ifp=1thenbp=bp-40:goto1980
1950 ifp=2thenbp=bp-1:goto1980
1960 ifp=3thenbp=bp-41:goto1980
1970 stop
1980 p=peek(bp)+peek(bp+1)+peek(bp+40)+peek(bp+41):ifp=4*32then2030
1990 gosub2480:goto1730
2000 rem" ┌──────────────────────────┐
2010 rem" | Spiel zu ende            |
2020 rem" └──────────────────────────┘
2030 poke53272,20:print"";
2040 printtab(11)" "
2050 printtab(11)" ┌──────────┐ "
2060 printtab(11)" |ame over| "
2070 printtab(11)" └──────────┘ "
2080 printtab(11)" "
2090 print"ewonnen hat:"
2100 fort=1to500:next
2110 ifq1=q2thenprint"unentschieden!":goto2160
2120 a$="spieler 1":ifq2>q1thena$="spieler 2"
2130 fori=1to200:print""a$:print""a$:next
2140 print"spieler 1:";q1
2150 print"spieler 2:";q2
2160 y1=y1+q1:y2=y2+q2:q1=0:q2=0
2170 print"Punktestand: spieler 1 hat"y1"Punkte"
2180 print"              spieler 2 hat"y2"Punkte"
2190 print"space fuer neues spiel!"
2200 geta$:ifa$<>" "then2200
2210 print"":goto1410
2220 rem" ┌──────────────────────────┐
2230 rem" | Sound 1                  |
2240 rem" └──────────────────────────┘
2250 pokesi+24,15:pokesi+4,0:pokesi+5,15:pokesi+6,255:pokesi+4,33
2260 fori=64to128step16:pokesi+1,255-i:next
2270 fori=64to128step16:pokesi+1,i:next
2280 pokesi+4,0
2290 sp=bp-1024-int((bp-1024)/40)*40:ifsp>20thenq1=q1+1:goto2310
2300 q2=q2+1
2310 return
2320 rem" ┌──────────────────────────┐
2330 rem" | Sound 2                  |
2340 rem" └──────────────────────────┘
2350 pokesi+4,0:pokesi+5,15:pokesi+6,255:pokesi+4,17
2360 fori=1to30step2:pokesi+1,30-i:next
2370 fori=1to30step2:pokesi+1,i:next
2380 pokesi+4,0:return
2390 rem" ┌──────────────────────────┐
2400 rem" | Sound 3                  |
2410 rem" └──────────────────────────┘
2420 pokesi+4,0:pokesi+5,15:pokesi+6,255:pokesi+4,33
2430 fori=1to40step2:pokesi+1,60-i:next
2440 pokesi+4,0:return
2450 rem" ┌──────────────────────────┐
2460 rem" | Sound 4                  |
2470 rem" └──────────────────────────┘
2480 pokesi+4,0:pokesi+5,40:pokesi+6,160:pokesi+4,129
2490 fori=1to10step.1:pokesi+1,10-i:next
2500 pokesi+4,0:return
2510 rem" ┌──────────────────────────┐
2520 rem" | MC-Code einlesen         |
2530 rem" └──────────────────────────┘
```

```
2540 fori=32768to33002:readj:Pokei,j:next
2550 data 169,155,133,63,169,5,133,64,169,172,133,65,169,5,133,66,32,20,128
2560 data 96,169,7,160,0,145,63,145,65,160,40,145,63,145,65,160,80,145,63
2570 data 145,65,96,169,32,160,0,145,63,145,65,160,40,145,63,145,65,160,80
2580 data 145,63,145,65,96,169,224,141,2,220,173,1,220,141,60,3,173,0,220,141
2590 data 61,3,169,255,141,2,220,32,41,128,173,60,3,41,1,240,25,173,60,3,41
2600 data 2,240,47,173,61,3,41,1,240,69,173:61,3,41,2,240,91,32,20,128,96,234
2610 data 165,64,201,4,208,6,165,63,201,91,240,225,56,165,63,233,40,133,63
2620 data 165,64,233,0,133,64,76,101,128,234,165,64,201,7,208,6,165,63,201
2630 data 43,240,196,24,165,63,105,40,133,63,165,64,105,0,133,64,76,101,128
2640 data 234,165,66,201,4,208,6,165,65,201,108,240,181,56,165,65,233,40,133
2650 data 65,165,66,233,0,133,66,76,115,128,234,165,66,201,7,208,6,165,65,201
2660 data 60,240,152,24,165,65,105,40,133,65,165,66,105,0,133,66,76,115,128
2670 rem" ┌──────────────────────────┐
2680 rem"| Sonderzeichen einlesen    |
2690 rem" └──────────────────────────┘
2700 fori=12544toi+7:Pokei,0:next
2710 fori=12288to12367:readj:Pokei,j:next
2720 data 180,252,252,104,120,120,88,120,127,111,127,126,92,124,124,124,45,63
2730 data 63,26,30,30,22,30,254,250,254,126,54,62,62,62,0,24,24,60,60,126,126
2740 data 24,255,255,255,255,239,199,131,1,255,255,170,85,170,85,255,255,235
2750 data 215,235,215,235,215,235,215,235,215,170,85,170,85,235,215,60,102,223
2760 data 191,191,255,126,60
2770 rem" ┌──────────────────────────┐
2780 rem"| Sprite-Datas              |
2790 rem" └──────────────────────────┘
2800 fori=0toi+7*64-1:readj:Poke200*64+i,j:next:return
2810 Poke53269,255:Poke53276,255:Poke53286,1:Poke53280,0:Poke53281,0
2820 Poke53248+23,255
2830 fori=0to7
2840 Poke53248+i*2,40+i*23
2850 Poke53249+i*2,50
2860 Poke2040+i,200+i+(i>5)
2870 Poke53287+i,14
2880 next
2890 Poke2045,202:Poke53265,27
2900 fori=0to7
2910 P=53287+i
2920 PokeP,1
2930 fort=1to10:next
2940 PokeP,14
2950 fort=1to10:next
2960 next
2970 geta$:z=z+1:ifz<50anda$=""then2900
2980 Poke53269,0
2990 return
3000 data 255,15,240,235,14,176,235,14,176,235,14,176,235,14,176,235
3010 data 14,176,235,14,176,235,254,176,235,174,176,235,174,176,235,174
3020 data 176,235,174,176,235,174,176,235,174,176,235,174,176,235,174,176
3030 data 235,174,176,234,170,176,234,170,176,234,170,176,255,255,240,0
3040 data 15,255,0,58,170,192,234,170,176,234,170,176,235,254,176,235
3050 data 14,176,235,14,176,235,14,176,235,254,176,234,170,176,234,170
3060 data 176,234,170,176,235,254,176,235,14,176,235,14,176,235,14,176
3070 data 235,14,176,235,14,176,235,14,176,235,14,176,255,15,240,0
3080 data 255,255,0,234,170,192,234,170,176,234,170,176,235,250,176,235
3090 data 14,176,235,14,176,235,14,176,235,250,176,234,170,176,234,170
3100 data 176,234,170,192,235,175,0,235,172,0,235,235,0,235,235,0
3110 data 235,58,192,235,58,192,235,14,176,235,14,176,255,15,240,0
3120 data 255,0,0,235,0,0,235,0,0,235,0,0,235,0,0,235
3130 data 0,0,235,0,0,235,0,0,235,0,0,235,0,0,235,0
3140 data 0,235,0,0,235,0,0,235,0,0,235,0,0,235,0,0
3150 data 235,255,240,234,170,176,234,170,176,234,170,176,255,255,240,0
3160 data 15,255,0,58,170,192,234,170,176,234,170,176,234,250,176,235
3170 data 14,176,235,14,176,235,14,176,235,14,176,235,14,176,235,14
3180 data 176,235,14,176,235,14,176,235,14,176,235,14,176,235,14,176
3190 data 234,250,176,234,170,176,234,170,176,58,170,192,15,255,0,0
3200 data 255,255,0,234,170,192,234,170,176,234,170,176,235,250,176,235
3210 data 14,176,235,14,176,235,14,176,235,14,176,235,14,176,235,14
3220 data 176,235,14,176,235,14,176,235,14,176,235,14,176,235,14,176
3230 data 235,250,176,234,170,176,234,170,176,234,170,192,255,255,0,0
3240 data 15,255,0,58,170,192,234,170,176,234,170,176,234,250,176,235
3250 data 14,176,235,15,240,235,0,0,234,255,0,234,170,192,234,170
3260 data 176,58,170,176,15,250,176,0,14,176,255,14,176,235,14,176
3270 data 234,250,176,234,170,176,234,170,176,58,170,192,15,255,0,0
```

PIPELINE
für Laser 2001

Oily Jake ist ein fleißiger Arbeiter in einer Ölraffinerie. Seine Aufgabe ist Ölleitungen zu überwachen, auftretende Schäden zu reparieren und darauf zu achten, daß die Ölfässer nicht überfüllt werden. Die Arbeit in der Ölraffinerie könnte richtig Spaß machen, wenn sich nicht ein giftgrüner Skorpion in die Raffinerie eingeschlichen hätte, der am laufenden Band versucht, Oily Jake's Arbeit zu sabotieren. Ferner muß Oily Jake, um seine Arbeit verrichten zu können, in schwindelerregender Höhe seinen Kontrollgang über die Pipelines durchführen. Dabei muß er aufpassen, daß er nicht abstürzt, und somit eines seiner 3 Leben verliert.

PIPELINE ist auf dem LASER 2001 geschrieben und wird mittels des linken Joysticks gesteuert.

Das Programm ist in 2 Teile gegliedert. Teil 1 ist mit „PIPELINE-VORPROGRAMM" gekennzeichnet, und belegt die komplette Grafik, sowie die 4 verschiedenen Grafikfenster im VideoRAM. Das Programm läuft ca. 3 Minuten. Die belegte Grafik bleibt so lange im VideoRAM, bis der Rechner ausgeschaltet wird, oder der „GR"-Befehl gegeben wird. Das Vorprogramm muß deshalb nur einmal vor dem ersten Spielbeginn laufen. Danach wird dieses Programm durch den 2. Teil „PIPELINE-HAUPTPROGRAMM" überschrieben. Dieser 2. Teil beinhaltet die komplette Spielhandlung sowie die Steuerung der 4 Grafikfenster und die Maschinenroutine zur Videoprozessorsteuerung.

Es ist wichtig, beide Programmteile vor dem ersten Start auf Kassette oder Diskette wegzuschreiben, da Teil 1 sofort nach dem Programmlauf Teil 2 einliest und startet und Teil 2 nach dem ersten Start nicht mehr editiert werden kann, ohne daß Programmteile zerstört werden.

Teil 2 darf nach einem Abbruch mit Break (CTRL C) nicht mehr mit RUN gestartet werden, sondern muß mit der Befehlsfolge CLEAR (Return) GOTO 12 (Return) neu gestartet werden.

Diskettenbenutzer sollten im ersten Programmteil die Zeile 100 umändern in: 100 DISK „RUN PIPELINE-HP" und den ersten Teil unter dem Namen PIPELINE sichern. Danach wird der 2. Teil unter dem Namen PIPELINE-HP auf die gleiche Diskette weggeschrieben. Das Programm wird nun mit RUN PIPELINE gestartet, und lädt dann automatisch den 2. Teil nach dem Programmende ein.

PIPELINE bringt einige Besonderheiten des LASER 2001 zutage, welche den meisten Besitzern unbekannt sein dürften:

Das Programm benutzt den Grafikmodus 1, der vom BASIC nicht erreicht werden kann, jedoch eine schnelle Farbgrafik, sowie die Mischung von Grafik und Text erlaubt.

Es werden 4 Grafikfenster benutzt, welche außerhalb des TEXT-Bereichs liegen. Somit wird der blitzschnelle Wechsel der einzelnen Bilder erreicht. Das Programm benutzt SPRITES, wozu der LASER 2001 normalerweise keinen BASIC-Befehl kennt. Damit wird die umfangreiche Bewegung, die Geschwindigkeit der bewegten Objekte, sowie die Grafiküberlagerung erreicht.

Doch nun genug der Vorbetrachtungen. Kommen wir zum eigentlichen Spiel: Sobald das Titelbild auf dem Bildschirm erschienen ist und die Titelmelodie verklungen ist, kann durch Druck auf den linken Aktionsknopf des linken Joysticks das Spiel begonnen werden. Jetzt erscheint die erste Pipeline auf dem Bildschirm.

Oily Jake steht auf dem linken oberen Ende der Pipeline. Rechts neben der Leiter wird ein Zeitband angezeigt, welches im Verlauf des Spieles immer kürzer wird. Sobald das Zeitband nur noch 32 Zeiteinheiten beinhaltet, ertönt ein Warnsignal, und das Band verfärbt sich rot.

Auf den unteren Sprossen der Leiter erscheint der giftgrüne Skorpion, welcher Oily Jake das Leben schwer macht. Der Level beginnt, sobald die PIPELINE-Fanfare verklungen ist.

Steuern Sie mit dem Joystick Oily Jake vertikal, horizontal oder diagonal. Achten Sie jedoch darauf, daß Oily Jake nicht von der Pipeline abkommt, da er sonst abstürzt und eines seiner 3 Leben verliert.

Das Ziel ist es, zum Ölfaß zu kommen, bevor das Zeitband abgelaufen ist. Ferner darf der Ölfluß nicht durch die Sabotage des Skorpions unterbrochen sein, wenn Oily Jake das Ölfaß erreicht. Sollten Sie das schaffen, so geht das Spiel mit einer anderen Pipeline und dem nächsthöheren Level weiter.

Insgesamt gibt es 3 verschiedene Pipelines und 8 Levels.

Der Skorpion klettert immer die Leiter hinauf, das Gerüst entlang und dann irgendwo über die Pipeline geradlinig zum unteren Bildschirmrand. Wenn Oily Jake vom Skorpion berührt wird, so stürzt er ab und stirbt eines seiner 3 Leben. Berührt der Skorpion eine Flansch der Pipeline, so geht die Pipeline an dieser Stelle kaputt, und der Ölfluß wird unterbrochen. Dann muß Oily Jake an die defekte Stelle geführt werden, um die Pipeline zu reparieren. Oily Jake hat jedoch gegen den Skorpion eine Waffe: Er kann den Skorpion mittels seines Hammers zu Fall bringen. Zu diesem Zweck muß Oily Jake mit dem Gesicht zum Skorpion stehen und der Aktionsknopf muß gedrückt werden. Dann fliegt der Hammer in Richtung Skorpion, sofern sich der Skorpion in vertikaler Richtung bewegt, wenn der Hammer den Skorpion trifft, so stürzt dieser ab, und beginnt wieder von neuem seine Klettertour auf der Leiter.

Oily Jake verliert eines seiner 3 Leben, wenn er:

1. von der Pipeline abkommt.
2. zum Ölfaß kommt, wenn die Pipeline defekt, und somit der Ölfluß unterbrochen ist.
3. wenn das Zeitband abgelaufen ist.

Punktwertung:

5 * Levelpunkte
für jeden Skorpion, der durch Hammerwurf zu Fall gebracht wurde.

500 * Levelpunkte
bei erfolgreichem Erreichen des Ölfasses.

2 * Levelpunkte
für jede verbliebene Zeiteinheit des Zeitbandes bei Erreichen des Ölfasses.

Schwierigkeitsstufen:

Das Spiel besteht aus maximal 8 Levels. Die Schwierigkeitssteigerung besteht darin, daß bei steigendem Level das Zeitband immer kürzer wird, und der Skorpion immer genauer mit dem Hammer getroffen werden muß. Bei einer Überschreitung von Level 8 geht das Spiel mit gleichbleibendem Level 8 weiter.

Wenn Oily Jake seine 3 Leben verloren hat, so erscheint wieder das Titelbild, und ein neues Spiel kann mittels Druck auf den Aktionsknopf gestartet werden.

```
1  REM *****************************
2  REM *  PIPELINE  VORPROGRAMM   *
3  REM * fuer LASER 2001+Joystick *
4  REM *--------------------------*
6  REM *  (c) 1984 by Tronicsoft  *
7  REM *  all Rights by V.Becker  *
8  REM *****************************
9  REM
10  REM
12  HOME : PRINT : PRINT  SPC( 10)"P I P E L I N E": PRINT  SPC( 10)"-----------
----": PRINT : PRINT
14  PRINT "   Grafik umbelegen......";
20  FOR I = 32 TO 218:A =  RAM (C$): READ C$: FOR K = 0 TO 7
30 G$ =  MID$ (C$,2 * K + 1,2):H$ =  LEFT$ (G$,1):L$ =  RIGHT$ (G$,1):R$ = H$: G
OSUB 200:H = R:R$ = L$: GOSUB 200:L = R
40  VPOKE 34815 + 8 * I + K,16 * H + L
50  NEXT K: NEXT I
55  PRINT "beendet"
56  PRINT "   Fenster belegen.......";
60  FOR I = 36607 TO 36766: VPOKE I,0: NEXT I
90  GOSUB 10000
95  GOSUB 11000
96  GOSUB 12000
97  GOSUB 13000
98  PRINT "beendet"
99  PRINT "   Hauptprogramm laden..."
100  CRUN
200  IF  ASC (R$) > 57 THEN R =  ASC (R$) - 55: RETURN
210 R =  VAL (R$): RETURN
10000  FOR I = 40959 TO 44031 STEP 1024
10005  FOR K = 0 TO 767: VPOKE I + K,32: NEXT K
10010  FOR K = 0 TO 736 STEP 32: VPOKE I + K,218: VPOKE I + K + 1,218: VPOKE I +
 K + 30,218: VPOKE I + K + 31,218: NEXT K
10015  FOR K = 4 TO 23: VPOKE I + 32 * K + 29,220 + K: NEXT K
10020  FOR K = 98 TO 125: VPOKE I + K,39: NEXT K
10030  FOR K = 155 TO 731 STEP 96: VPOKE I + K,37: NEXT K
10040  FOR K = 156 TO 732 STEP 96: VPOKE I + K,38: NEXT K
10050  FOR K = 187 TO 763 STEP 96: VPOKE I + K,33: NEXT K
10060  FOR K = 188 TO 764 STEP 96: VPOKE I + K,34: NEXT K
10070  FOR K = 219 TO 699 STEP 96: VPOKE I + K,35: NEXT K
10080  FOR K = 220 TO 700 STEP 96: VPOKE I + K,36: NEXT K
10090  FOR K = 3 TO 5: VPOKE I + K,K + 55: NEXT K
10100  FOR K = 25 TO 27: VPOKE I + K,K + 36: NEXT K
10110  VPOKE I + 725,40: VPOKE I + 726,42: VPOKE I + 757,41: VPOKE I + 758,43
10120  NEXT I
10125  FOR I = 33535 TO 33662: VPOKE I,0: NEXT I
10126  FOR I = 41349 TO 41364: READ C: VPOKE I,C: VPOKE I + 32,C + 1: NEXT I
10127  VPOKE 33555,175: VPOKE 33556,168: VPOKE 33557,44: VPOKE 33558,7
10128  FOR I = 41447 TO 41456: VPOKE I,I - 41239: NEXT I
10130  RETURN
11000  FOR I = 4 TO 30: READ C: VPOKE I + 32831,C: NEXT I: RETURN
12000  FOR I = 41983 TO 44031 STEP 1024
12010  FOR J = 6 TO 21: READ C$
12020  FOR K = 1 TO  LEN (C$): VPOKE I + 32 * J + 1 + K, ASC ( MID$ (C$,K,1)): N
EXT K
12025 X =  RAM (C$)
12030  NEXT J: NEXT I
12040  RETURN
13000 I = 40959
13010  FOR J = 6 TO 7:X =  RAM (C$): READ C$
13020  FOR K = 1 TO  LEN (C$): VPOKE I + 32 * J + 1 + K, ASC ( MID$ (C$,K,1)): N
EXT K
13030  NEXT J
13040  FOR J = 8 TO 15:X =  RAM (C$): READ C$
13050  FOR K = 1 TO  LEN (C$): VPOKE I + 32 * J + 24 + K, ASC ( MID$ (C$,K,1)):
NEXT K
13060  NEXT J
13070  FOR J = 16 TO 20:X =  RAM (C$): READ C$
13080  FOR K = 1 TO  LEN (C$): VPOKE I + 32 * J + 20 + K, ASC ( MID$ (C$,K,1)):
NEXT K
13090  NEXT J
13100  RETURN
60000  DATA "0000000000000000","FF2020FF2020FF20","FF0404FF0404FF04","20FF2020FF
2020FF","04FF0404FF0404FF","2020FF2020FF2020","0404FF0404FF0404","FFFF3C7EE7C3FF
FF"
```

```
60010  DATA "001F103F7F3F1010","1010103F7F3F101F","00F808FCFEFC0808","080808FCFE
FC08F8","000000000000000E","0A0E000000000000","00000000000000A0","A0B0000000000
00"
60020  DATA "0018242424241800","0008180808080800","001824040810300","0018240804
241800","0028282838080800","003C203804241800","0018203824241800","003C0408101010
00"
60030  DATA "0018241824241800","001824241C041800","00F08087F414F700","000000BD25
25BD00","0000005E9E101E00","008080878784F700","000000A2A2148800","000404F4F484F6
00"
60040  DATA "183C3CFFFF3C3C3C","3C3C3CFFFF3C3C18","18187FFFFF7F1818","1818FEFFFF
FE1818","010303FFFF030303","030303FFFF030301","80C0C0FFFFC0C0C0","C0C0C0FFFFC0C0
80"
60050  DATA "000000FFFF000000","0000000000000000","1818181818181818","0000000000
000000","18181818187FFF","1818181818FEFF","FF7F181818181818","FFFE181818181818
18"
60060  DATA "0307070000000007","0707070707000000","0080E00000000080","C0C0C0C080
000000","0001070000000001","0303030301000000","C0E0E000000000E0","E0E0E0E000000
00"
60070  DATA "0000000000000000","1C0F030000000000","0000E0FF1F000000","0038202020
202038","0000000001202875","FFFF7D0000000000","000000004080F1FF","F8F9FF0000000
00"
60080  DATA "0000201008000000","000000000030F1C","0000001FFFE00000","0000000103
030606","3060C08000000001","000000071F78E080","0C0C0C1818181818","0306060C0C1818
18"
60090  DATA "000000F8FF070000","00000000C0F038","000000E0F81E0701","0C06030100
000080","00000080C0C06060","C060603030181818","3030301818181818","18181830306060
C0"
60100  DATA "1818181818303030","01071EF8E0000000","800000000103060C","6060C0C080
000000","000007FFF8000000","38F0C00000000000","18181818180C0C0C","1818180C0C0606
03"
60110  DATA "0606030301000000","0100000080C06030","80E0781F07000000","0018202040
202018","01030305050A0A12","1200000000000000","80C0C0A0A0505048","48000000000000
00"
60120  DATA "0000000F07070300","0000000007040406","0000008040800408","1420000080
8080C0","0000000F07070300","0000000000040406","0000008040800004","38040000004028
10"
60130  DATA "0000000102012010","2804000001010103","000000F0E0E0C000","00000000E0
202060","0000000102010020","1C20000000021408","000000F0E0E0C000","00000000002020
60"
60140  DATA "0000000001030701","0101010000000000","00000000080E0E080","8080800000
000000","0000000001070701","0101010000000000","00000000080C0E080","80808000000000
00"
60150  DATA "0000000000000007","0700000000000000","00000000004060F0","F060600000
000000","000000000002060F","0F06060000000000","00000000000000E0","E000000000000
00"
60160  DATA "1436361409050303","0303030100000000","286C6C2890A0C0C0","C0C0C08080
502000","081C1C0809050303","03030301010A0400","1038381090A0C0C0","C0C0C080000000
00"
60170  DATA "000060F008F4630F","0F63F408F0600000","000000000402E4F8","F0E0000000
000000","00000060F864030F","0F0364F840000000","00000000000E0F0","F8E40204000000
00"
60180  DATA "00040A0101030303","0303050914363614","00000000080C0C0C0","C0C0A09028
6C6C28","0000000001030303","03030509081C1C08","0020508080C0C0C0","C0C0A090103838
10"
60190  DATA "3F7FFFE0E0E0E0FF","FFFFE0E0E0C080","E0F0F078783030E0","C080000000
000000","0F1F3F0707070707","070707070F1F3F","F8F0E08080808080","80808080F8F0
E0"
60200  DATA  "3F7FFFE0E0E0E0FF","FEFCE0E0E0FFFFFF","F8F0E00000000000","000000000
0F8F0E0","2060E0E0E0E0E0E0","E0E0E0E0E0FFFFFF","0000000000000000","000000000F8F
0E0"
60210  DATA "2060E0F8FCFEFFEF","E7E7E3E1E0E0C080","08183838383838B8","B8F8F8F8F8
783020","3038702030383048","0000000000000000","0000000000000000","00000000000000
00"
60220  DATA "0003050606050300","00C4AC2424A4C400","00F79497F414F700","00D252D25F
42C200","003F080A0B0A0A00","00FF00DE12121E00","00F500B5D5959500","00CF08C80F00CF
00"
60230  DATA  "00FE013DA5A5BD00","00D010B810101000","FFFFFFFFFFFFFFFF"
60235  DATA 184,186,188,190,184,186,192,194,196,198,188,190,200,202,192,194
60240  DATA  129,161,241,241,81,81,81,81,81,81,81,81,0,0,0,0,0,0,0,33,33,33,209,
209,161,161,161
60250  DATA "HHHHHHHHHHHHHHHHHHH@hi","HHHHHHHHHHHHHHHHHHHAjkl","
     mn","                              BC","                              oP"
60260  DATA "    ab@HHHHHHHHHHHHHH@qrs","  cdeAHHHHHHHHHHHHHHAtu","  f9"
60265  DATA "  BC","  JJ            abDFhi","  JJ            cdeEGjkl"
60270  DATA "  JJ          f9    mn","  BC            BC    BC","  vw              oP
JJ","  xyz@HHHHH@qrs    BC","   YZAHHHHHAtu"
```

```
60280  DATA "HHHH@hi     ab@HHHHHH@hi","HHHHAjkl  cdeAHHHHHHAjkl","        mn  f9
     mn","      BC  BC            LM","    oP  JJ           NO"
60290  DATA  "  ab@qrs  JJ              oP"," cdeAtu   JJ   ab@HH@qrs"," f9          JJ
  cdeAHHAtu"," BC        JJ  f9"," JJ        JJ  LM"," JJ         JJ  NO"
60300  DATA " JJ        JJ  vw"," BC         BC  xyz@hi"," vw        oP   YZAjkl","
xyz@HHH@qrs        mn","  YZAHHHAtu"
60310  DATA "HHHHHHHH@hi","HHHHHHHHAjkl","              mn   ab@HH@hi","           BC
  cdeAHHAjkl","            JJ  f9      mn","             JJ  BC      BC"
60320  DATA "           BC  JJ        JJ","             oP  JJ      JJ","  ab@HH@qrs
 JJ       JJ","  cdeAHHAtu   JJ        JJ","  f9            JJ       JJ"
60330  DATA "  LM            JJ        BC","  NO              BC         oP","  vw
 oP   ab9rs","  xyz@HHHHH@qrs   Jetu","   YZAHHHHHHAtu"
60340  DATA "HHHH@HHHHHHHHHHHHHHHH@hi","HHHHAHHHHHHHHHHHHHHHHHAjkl","mn","BC","JJ
","JJ","JJ","JJ","BC","oP"," abqrs","cdetu","f9","BC","JJ"




1  REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
2  REM ***************************
3  REM * PIPELINE - HAUPTPROGRAMM *
4  REM * fuer LASER 2001+Joystick *
5  REM *-------------------------*
6  REM *  (c) 1984 by Tronicsoft  *
7  REM *  all Rights by V.Becker  *
8  REM ***************************
9  REM
12  DIM C(2)
15  FOR I = 59968 TO 59988: POKE I - 39482, PEEK (I): NEXT I
20  POKE 20489,19: POKE 20490,80: POKE 20497,96: POKE 20498,234
30  POKE 20499,0: POKE 20500,226: POKE 20501,8: POKE 20502,1: POKE 20503,1: POKE
 20504,6: POKE 20505,1: POKE 20506,13
40 BA = 40959:W = 0: GOSUB 60000
45  RESTORE
47  FOR I = 1 TO 12: READ S: NEXT I: GOSUB 61000: GOSUB 61100
50 LB = 3:LV = 1:ST = 0
55 W = 1
60  VPOKE 33535,8: VPOKE 33536,208: VPOKE 33537,84: VPOKE 33538,7: VPOKE 33539,8
: VPOKE 33540,208: VPOKE 33541,136: VPOKE 33542,11: VPOKE 33543,8: VPOKE 33544,2
24: VPOKE 33545,168: VPOKE 33546,2
62  VPOKE 33567,165: VPOKE 33568,168: VPOKE 33569,124: VPOKE 33570,1
65  VPOKE 33543,8: VPOKE 33544,224: VPOKE 33545,168: VPOKE 33537,84
70  FOR I = 208 TO 16 STEP  - 1: VPOKE 33536,I: VPOKE 33540,I: VPOKE 33544,I + 1
6
80  IF  NOT ((I / 2) -  INT (I / 2)) THEN  VPOKE 33541,136: VPOKE 33545,168: VPO
KE 33570,1: GOTO 95
90  VPOKE 33541,140: VPOKE 33545,172: VPOKE 33570,8
95  CALL  - 1536: IF  PEEK (24) > 100 THEN 400
100  NEXT I
105  VPOKE 33543,232: VPOKE 33544,216: VPOKE 33545,160: VPOKE 33537,80
110  FOR I = 16 TO 208: VPOKE 33536,I: VPOKE 33540,I: VPOKE 33543,232 - I
120  IF  NOT ((I / 2) -  INT (I / 2)) THEN  VPOKE 33541,128: VPOKE 33545,160: VP
OKE 33570,1: GOTO 195
130  VPOKE 33541,132: VPOKE 33545,164: VPOKE 33570,8
195  CALL  - 1536: IF  PEEK (24) > 100 THEN 400
200  NEXT I
210  GOTO 65
400  GOSUB 60300:P = 0
500  FOR I = 33535 TO 33546: VPOKE I,0: NEXT I: VPOKE 33570,1
510  GOSUB 60000
515 ZM = 34:SM = 16:SD = 1:ZK = 176:SK = 216:MW = 1:SL = 0
520  RESTORE
525  VPOKE 32861,161: FOR ZB = 36766 TO 36607 + (8 * (LV - 1)) STEP  - 1: VPOKE
ZB,60: NEXT ZB
530  FOR I = 33535 TO 33546: READ S: VPOKE I,S: NEXT I: GOSUB 61000
535  FOR LP = 1 TO 2
540  CALL  - 1536
545  IF  PEEK (24) > 100 THEN  GOSUB 40000: CALL  - 1536
550 JS =  PEEK (24) - ( PEEK (24) > 100) * 128
560  IF  NOT JS THEN 590
570 JS = JS / 2: IF JS > 10 THEN 5000
580  ON JS GOSUB 10000,10100,10200,10300,10800,10400
585  GOTO 600
```

```
590  FOR DL = 1 TO 90: NEXT DL
595  ON MW GOSUB 27000,27100,27200,27300,27400,27500,28000
600  ON LP GOSUB 25000,25100
605  NEXT LP
610  IF ST THEN 30000
615  GOTO 535
5000 JS = JS - 15
5010  ON JS GOSUB 10500,10600,10800,10800,10700
5020  GOTO 600
10000 ZM = ZM + 2: ON SD GOSUB 21000,23000
10010  GOSUB 20100: GOSUB 20000:ZM = ZM + 2: ON SD GOSUB 22000,24000
10020  GOSUB 20100,20000: RETURN
10100 SM = SM + 2: GOSUB 21000: GOSUB 20100: GOSUB 20000:SM = SM + 2: GOSUB 2200
0: GOSUB 20100: GOSUB 20000: RETURN
10200 ZM = ZM + 2:SM = SM + 2: GOSUB 21000: GOSUB 20100: GOSUB 20000:ZM = ZM + 2
:SM = SM + 2: GOSUB 22000: GOSUB 20100: GOSUB 20000: RETURN
10300 ZM = ZM - 2: ON SD GOSUB 21000,23000
10310  GOSUB 20100: GOSUB 20000:ZM = ZM - 2: ON SD GOSUB 22000,24000
10320  GOSUB 20100,20000: RETURN
10400 ZM = ZM - 2:SM = SM + 2: GOSUB 21000: GOSUB 20100: GOSUB 20000:ZM = ZM - 2
:SM = SM + 2: GOSUB 22000: GOSUB 20100: GOSUB 20000: RETURN
10500 SM = SM - 2: GOSUB 23000: GOSUB 20100: GOSUB 20000:SM = SM - 2: GOSUB 2400
0: GOSUB 20100: GOSUB 20000: RETURN
10600 ZM = ZM + 2:SM = SM - 2: GOSUB 23000: GOSUB 20100: GOSUB 20000:ZM = ZM + 2
:SM = SM - 2: GOSUB 24000: GOSUB 20100: GOSUB 20000: RETURN
10700 ZM = ZM - 2:SM = SM - 2: GOSUB 23000: GOSUB 20100: GOSUB 20000:ZM = ZM - 2
:SM = SM - 2: GOSUB 24000: GOSUB 20100: GOSUB 20000: RETURN
10800  RETURN
20000  VPOKE 33535,ZM: VPOKE 33536,SM: VPOKE 33539,ZM: VPOKE 33540,SM: GOSUB 260
00
20010  ON MW GOSUB 27000,27100,27200,27300,27400,27500,28000
20020  RETURN
20100 SM = (SM >  = 216) * 216 + (SM <  = 16) * 16 + ((SM > 16) AND (SM < 216))
* SM
20200 ZM = (ZM >  = 176) * 176 + (ZM <  = 34) * 34 + ((ZM > 34) AND (ZM < 176))
* ZM: RETURN
21000 SD = 1: VPOKE 33537,80: VPOKE 33541,132: SOUND (165,1,15): RETURN
22000  VPOKE 33537,80: VPOKE 33541,128: SOUND (148,1,15): RETURN
23000 SD = 2: VPOKE 33537,84: VPOKE 33541,140: SOUND (165,1,15): RETURN
24000  VPOKE 33537,84: VPOKE 33541,136: SOUND (148,1,15): RETURN
25000  IF SL THEN  RETURN
25010  VPOKE 33570,8: RETURN
25100  IF SL THEN 25110
25105  VPOKE 33570,1
25110  IF  NOT (ZB - 36767) THEN ST = 1
25115  IF ZB = 36735 THEN  VPOKE 32861,145: SOUND (195,2,15): SOUND (175,2,15):
SOUND (195,2,15)
25120  VPOKE ZB,0:ZB = ZB + 1: RETURN
26000 ST = 0:Z = 32 *  INT (ZM / 8 + 2):S =  INT (SM / 8): IF  VPEEK (WI + 2 + Z
 + S) = 32 THEN ST = 1
26010  RETURN
27000 ZK = ZK - 4: VPOKE 33543,ZK: VPOKE 33545,160: IF ZK <  = 24 THEN  VPOKE 33
543,8:MW = 4:EW =  INT ( RND (255) * 176) + 24: RETURN
27010 MW = 2: RETURN
27100 ZK = ZK - 4: VPOKE 33543,ZK: VPOKE 33545,164: IF ZK <  = 24 THEN  VPOKE 33
543,8:MW = 3:EW =  INT ( RND (255) * 176) + 24: RETURN
27110 MW = 1: RETURN
27200 SK = SK - 4: VPOKE 33544,SK: VPOKE 33545,168: IF SK <  = EW THEN :MW = 6:
RETURN
27210 MW = 4: RETURN
27300 SK = SK - 4: VPOKE 33544,SK: VPOKE 33545,172: IF SK <  = EW THEN :MW = 5:
RETURN
27310 MW = 3: RETURN
27400 ZK = ZK + 4: VPOKE 33543,ZK: VPOKE 33545,176: IF ZK >  = 176 THEN ZK = 176
:MW = 2:SK = 216: VPOKE 33544,SK: GOTO 27420
27410 MW = 6: GOSUB 27600
27420  RETURN
27500 ZK = ZK + 4: VPOKE 33543,ZK: VPOKE 33545,180: IF ZK >  = 176 THEN ZK = 176
:MW = 1:SK = 216: VPOKE 33544,SK: GOTO 27420
27510 MW = 5: GOSUB 27600: RETURN
27600 SL = 0:Z1 = 32 *  INT (ZK / 8 + 2):S1 =  INT (SK / 8):VG =  VPEEK (WI + 2
+ Z1 + S1)
27610  IF ((VG > 63) AND (VG < 72)) OR ((VG > 75) AND (VG < 80)) THEN SL = 1
27615  IF ( ABS (SM - SK) < 12) AND ( ABS (ZM - ZK) < 12) THEN ST = 1
```

```
27620  IF  NOT SL THEN  RETURN
27630 CP = WI + 1 + Z1 + S1: VPOKE CP,73: VPOKE 33546,0:ZK = 176:SK = 216: VPOKE
 33543,ZK: VPOKE 33544,SK:MW = 7: VPOKE 33570,1
27640  RETURN
28000 Z2 = 32 *  INT (ZM / 8 + 2):S2 =  INT (SM / 8): IF  VPEEK (WI + 2 + Z2 + S
2) = 73 THEN 28100
28010  RETURN
28100  ON SD GOTO 28110,28120
28110 C(1) = 144:C(2) = 152: GOTO 28130
28120 C(1) = 148:C(2) = 156
28130  VPOKE 33538,0
28140  FOR R = 1 TO 10: FOR I = 1 TO 2: VPOKE 33541,C(I): IF I / 2 =  INT (I / 2
) THEN  SOUND (135,3,15): GOTO 28160
28150  SOUND (16,3,15)
28160  NEXT I: NEXT R
28170  IF  NOT (SD - 1) THEN  VPOKE 33541,132: GOTO 28190
28180  VPOKE 33541,140
28190  VPOKE 33538,7:MW = 1:SL = 0: VPOKE CP,VG: VPOKE 33546,2: RETURN
30000  IF  ABS (ZM - 165) < 12 AND  ABS (SM - 168) < 16 THEN 31000
30005  FOR FL = ZM TO 181 STEP 2: VPOKE 33535,FL: VPOKE 33539,FL: SOUND (286 - F
L,1,15): NEXT FL
30010  VPOKE 33538,0: VPOKE 33541,92
30012  SOUND (255,1,0),(255,1,0),(255,1,0)
30014  SGEN 228
30016  FOR I = 240 TO 255: SGEN I: FOR D = 0 TO 75: NEXT D: NEXT I: GOSUB 60400
30020  IF  NOT SL THEN 30025
30022  VPOKE CP,VG
30025  FOR L = ZB TO 36766: VPOKE L,0: NEXT L
30030  IF  NOT LB THEN 40
30035 ST = 0: GOTO 500
31000  IF SL THEN 30005
31002 P = P + 500 * LV: GOSUB 60200: GOSUB 61100
31005 LV = LV + 1: IF LV > 8 THEN LV = 8
31010 W = W + 1: IF W > 3 THEN W = 1
31020  SGEN 228,255: FOR L = ZB TO 36766:P = P + 2 * LV: SGEN 190,240: GOSUB 602
00: SGEN 255: VPOKE L,0: NEXT L
31025  GOTO 30030
40000  IF MW = 3 OR MW = 4 THEN  RETURN
40002  IF SD = 1 AND SK < SM THEN  RETURN
40004  IF SD = 2 AND SK > SM THEN  RETURN
40006  IF SL THEN  RETURN
40010  VPOKE 33547,ZM: VPOKE 33548,SM: VPOKE 33549,148: VPOKE 33550,14
40020  FOR I = SM TO SK STEP 8 *  SGN (SK - SM)
40030  VPOKE 33548,I: ON SD GOSUB 40100,40200
40035  SOUND (135,1,15)
40040  VPOKE 33548,I + 4: ON SD GOSUB 40300,40400
40045  SOUND (16,1,15)
40050  NEXT I
40060  VPOKE 33550,0: IF  ABS (ZK - ZM) > (14 - LV) THEN  RETURN
40070  FOR I = ZK TO 185 STEP 2: VPOKE 33543,I: SOUND (186 - I,1,15): NEXT I
40080 SK = 216:ZK = 176: VPOKE 33543,ZK: VPOKE 33544,SK: VPOKE 33545,160:MW = 2
40090 P = P + 5 * LV: GOSUB 60200: RETURN
40100  VPOKE 33549,144: RETURN
40200  VPOKE 33549,148: RETURN
40300  VPOKE 33549,152: RETURN
40400  VPOKE 33549,156: RETURN
60000 WI = BA + W * 1024:WA = W + 8: GOSUB 60100: GOSUB 60200: POKE 20501,WA: CA
LL 20486: RETURN
60100  VPOKE 29 + WI,LV + 48: RETURN
60200 A =  RAM (P$):P$ =  STR$ (P): FOR PA = 1 TO  LEN (P$): VPOKE WI + 6 + PA,
ASC ( MID$ (P$,PA,1)): NEXT PA: RETURN
60300  FOR I = 0 TO 3:AD = 40959 + I * 1024: FOR K = 6 TO 20: VPOKE AD + K,32: N
EXT K
60310  FOR K = 17 TO 19: VPOKE AD + K,204: NEXT K: NEXT I: RETURN
60400  FOR I = 0 TO 3:AD = 40959 + I * 1024 + 16: VPOKE AD + LB,32: NEXT I:LB =
LB - 1: RETURN
61000  FOR I = 1 TO 18: READ L,G1,G2: SOUND (G1,L,15),(G2,L,15): NEXT I: RETURN
61100  FOR I = 1 TO 10: READ L,G1,G2: SOUND (G1,L,15),(G2,L,15): NEXT I: RETURN
62000  DATA  34,16,80,7,34,16,128,11,176,216,160,2
62010  DATA 6,175,160,6,160,135,3,165,148,3,160,135,6,148,128,6,160,135,6,135,11
3,3,148,128,3,135,113,6,128,95
62020  DATA  6,135,113,6,113,75,6,95,65,6,165,148,3,160,135,3,165,148,6,148,75,1
2,135,65
62030  DATA  3,184,165,3,175,160,6,165,148,3,175,160,3,165,148,6,160,135,3,165,1
48,3,160,135,6,148,75,12,135,65
```

Fortsetzung von Seite 11
ein Kabel unschön um das Gerät liegt. Das an sich schon klobige Modul des Floppies ragt weit rechts hinaus und bedeutet wiederum einen Platzverlust, zumal das Flachbandkabel zum Laufwerk sperrig ist und zwischen Eingabeeinheit und Floppy gelegt werden muß. Wäre diese Schnittstelle an der Geräterückseite angebracht, so könnte man das unschöne Kabel dezent an der Rückwand des Arbeitstisches verstekken.

Fazit:

– Der Dragon hat an Schnittstellen alles, was man zur kommerziellen Anwendung benötigt, es fehlt hier an nichts, nur hätten die Konstrukteure die Anordnung optimieren können, hätten sie die Interfaces an der Rückseite des Computers (wie bei anderen Systemen praktiziert) angebracht.

Platz genug wäre vorhanden, da im Gerät selbst sehr großzügig mit dem reichlich vorhandenen Raum umgegangen wurde.

(Dies ist aber nicht unbedingt ein Nachteil – siehe Kat. 4)

Die Anschlüsse wackeln nicht, was mir bei manch anderem System unangenehm aufgefallen ist – sie sind sicher vor Anstoßen und ähnlichen schädlichen Aktionen seitens der User angebracht.

Innerer Aufbau

Ausnutzung des Raumes:

– Öffnet man den Dragon, so fällt sofort die saubere und übersichtliche Anordnung der technischen Elemente auf.

Mit dem vorhandenen Platz wurde hier sehr großzügig umgegangen. Diese Tatsache erleichtert erheblich eine evtl. Reparatur, bzw. Einstellungen im Gerät.

Auch dem Laien bereitet es keine Schwierigkeiten, die wichtigen Teile wie CPU, diverse IC's, Regler u. dgl. zu finden.

Alle Teile sind sauber verlötet, bzw. fest verschraubt und leicht zugänglich.

Negativum:

– Durch den oben beschriebenen großzügigen Umgang mit dem vorhandenen Raum ist die Ausnutzung nicht optimal.

Basic-Interpreter

Microsoft-Extended-Colour-Basic:

– Im Dragon ist ein äußerst komfortables implementiert. Viele andere Systeme müssen eine extended Basic-Version erst einladen (Steckmodul), was Speicherplatzverlust zur Folge hat. Mancher Kassenschlager kann sich hier ein großes Stück abschneiden! Mit Graphikbefehlen wie z. B. Draw, Line, Paint, Color u.s.w. läßt sich relativ einfach in hochauflösender Graphik programmieren. Die Musikbefehle Play und Sound sind m. E. die einfachste Möglichkeit, Musik und Geräusche zu erzeugen – beim Dragon gibt es kein umständliches poken und peeken – daraus folgt:

Zeit- und Speicherplatzersparnis bei Arbeiten mit Graphik und Musik! Auch die Stringbehandlung (zur Textverarbeitung pp.) ist denkbar komfortabel mit Befehlen wie:

INSTR, STRS, MIDS, ASC, CHRS, VAL, u.s.w.

Mit 16K ROM hat der Dragon zwar ein durchschnittliches ROM in Bezug auf die Quantität, aber die Befehle sind von großer Komfortabilität – der User spart Speicherplatz und Zeit en masse.

Diskettenbetrieb:

– Wenn man eine Diskettenstation angeschlossen hat, so stehen, außer den für Diskettenbetrieb notwendigen Befehlen, noch weitere Befehle wie z. B. AUTO, BEEP, ON ERROR GOTO, ERL, ect.

Diese Befehlserweiterung kann den Drachen zum Apfelbeißer machen, von Konkurrenten in gleicher Preisklasse ganz zu schweigen.

Benutzerfreundlichkeit

Peripherie:

– Durch die Eingangs erwähnte professionelle Tastatur eignet sich der Dragon bestens für den PC Bereich.

Das Basic trägt noch seinen Teil dazu bei, ebenso die erhältliche Peripherie: Drucker (Matrixdrucker), sehr schnelles Diskettenlaufwerk (ausbaufähig), Farbmonitor, Datenrecorder mit hoher Aufzeichnungsgeschwindigkeit, 2 Joysticks (leichtgängig, reagieren sofort, Hebel bleibt in gewünschter Stellung, gute Handlichkeit, gut erreichbarer Druckknopf).

Betriebssystem:

– Das nur für den Dragon 64 erhältliche OS-9 Betriebssystem, das dem verbreiteten, aber teueren, UNIX-Betriebssystem ähnlich ist, hebt den Dragon 64 in die PC-Klasse hinein.

Für knapp 5000,– DM hat der Anwender ein vollwertiges, kommerziell nutzbares Betriebssystem mit Anwendersoftware (Spellcheck, Mailmerge, C-Compiler, ect.) und weiteren höheren Programmiersprachen. OS-9 ist nur in Verbindung mit einer Diskettenstation zu nutzen.

Software:

– Die erhältliche Originalsoftware für den Heimcomputerfreak ist trotz Verdoppelung des Angebotes an Spielen sehr spärlich. Außerdem ist mir aufgefallen, daß auch die Originalsoftware von Dragon nicht alle Möglichkeiten, die der Computer auf grafischem und auf musikalischem Gebiet bietet, ausnutzt.

So sind auch beim Dragon mehr als vier Farben pro Graphikmodus möglich – auch die Erstellung von Mischfarben – sowie durchaus anspruchsvolle Musikeffekte bis hin zum polyphonen Spiel!

Die Anwendung der Software wartet schon mit qualitativ besseren Produkten auf, wie z. b. Alldream Assembl./Disassembl., Forth, Graphik-Animator, Finanzverwaltung, Dateiverwaltung, Synther 7, Music Box, usw.

Zu erwähnen sind hier noch die mannigfaltigen Programme von englischen Usern (vornehmlich Anwendungsprogramme).

CPU:

– Die CPU 6809E besitzt mannigfaltige Adressierungsarten, die diesen Prozessor zu einer Einheit der gehobenen Klasse macht – sie arbeitet intern mit 16 Bit!

Der Prozessor verfügt über 59 verschiedene Befehle in Verbindung mit 10 Adressierungsarten.

Daraus ergibt sich eine totale Anzahl von 1464 Möglichkeiten.

Handbuch

– Das Handbuch zum Dragon stellt einen weiteren Lichtblick dar, denn es ist didaktisch gut aufgebaut, beinhaltet für den Einsteiger genügend Grundinformationen und ist grammatisch und orthographisch korrekt gestaltet.

Anfangs verwirrt die Aufteilung der durchgehenden Erklärungen und der zusammenfassenden Erläuterungen etwas, jedoch sind sie farblich voneinander abgegrenzt, so daß man recht schnell durchblickt.

Verschiedene Anhänge am Ende des Buches verdeutlichen die Struktur des Dragon, außerdem enthalten:

Print at-Arbeitsblatt, Arbeitsblatt für niedrig- und hochauflösende Graphik.

Vergleiche zu anderen Systemen

– In seiner Preisklasse hat der Dragon auf kommerziellem Gebiet, nach meinen Erfahrungen, auch mit anderen Systemen, keinen Konkurrenten, wohl aber im Bereich der Spiele – hier müssen sich die Programmierer noch etwas bemühen – vor allem ist ihnen anzuraten, die Möglichkeiten des Computers auszuschöpfen und nicht nur nach Schema F Programme nach Standardmuster zu entwerfen.

Für kommerzielle Anwendungen, ich denke hier insbesondere an Klein- und Mittelbetriebe, hat der Dragon 64 das günstigste Preis/Leistungsverhältnis.



Businessgrafik

Dragon 32

(–OS-9–, großer RAM-Speicher, Basic, Schnittstellen, u.s.w.)
Zur Textverarbeitung stehen 48K nutzbar für Basic zur Verfügung (mehr als ...); das sehr schnelle und durch große Speicherkapazität auffallende Diskettenlaufwerk ermöglicht ein bequemes Arbeiten – in diesem Zusammenhang wäre noch zu erwähnen, daß der Dragon 64

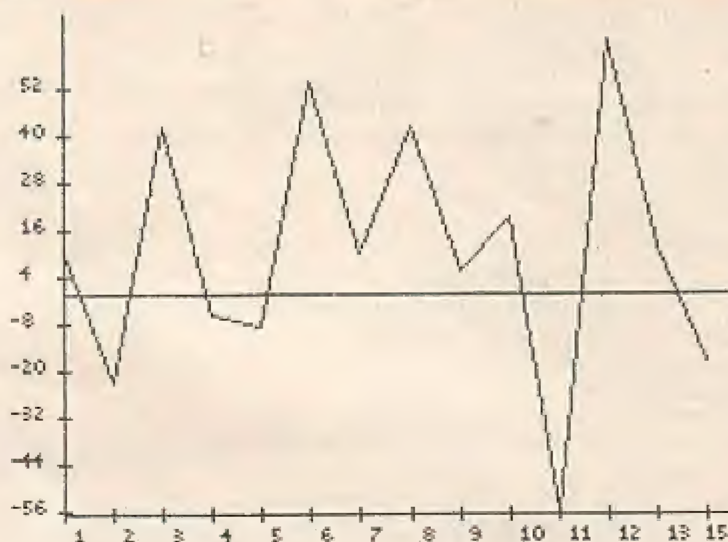
a) in Maschinensprache die vollen 64 K zur Verfügung stellt
b) softwaremäßig völlig kompatibel zu dem Dragon 32 ist und teilweise auch zu den anderen Rechnern mit 6809E CPU!

Das Diskettenlaufwerk ermöglicht den Direktstart eines Programmes – d. h. ein im Diskkatalog gespeichertes Programm kann mittels Eingabe von „RUN"+Name sofort zum Ablaufen gebracht werden.
(Kombinierte LOAD/RUN-Eingabe)

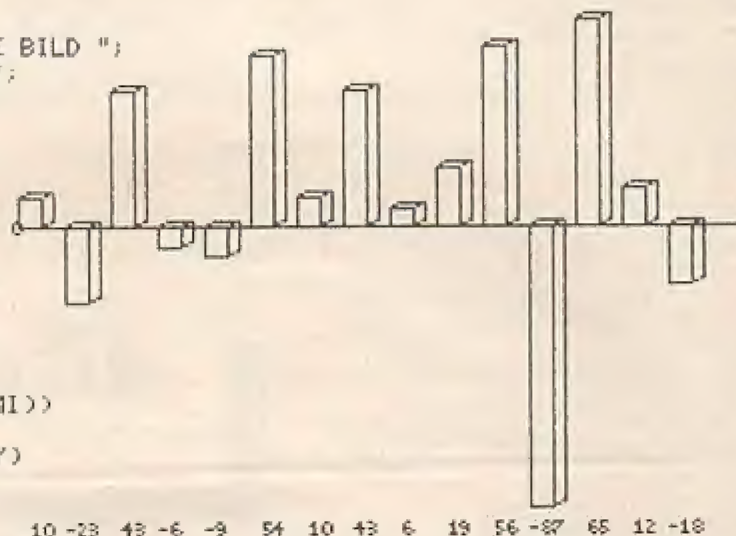
Ich würde, auch nach einigen Tests mit verschiedenen Systemen in der Preisklasse bis 1500,–, wieder auf das Dragon-System zurückgreifen.
Meiner Erfahrung nach ist der Dragon-Computer (insbesondere der Dragon 64) ein Gerät, das dem Anwender kaum Wünsche offen läßt, sei es in Maschinenprogrammierung (– CPU 6809E/intern 16Bit) oder in Basic (Extended colour Basic implementiert) oder in kommerzieller Ausnützung (OS-9 System, C-Compiler, Pascal, Forth).
Die noch offenstehenden Wünsche hinsichtlich guter Spiele dürften eigentlich kein Problem darstellen, zumal die User schon mit teilweise sehr guten Programmen aufwarten können!

```
10 CLEAR200,32498:CLS
20 REM***diagramme***
30 REM***(C)1984BY***
40 REM***a.woehrle***
50 REM*v.markmiller**
60 REM***************
70 PRINT@104,"businessgrafik"
80 PRINT@136,"=============="
90 PRINT@231,"(c) 1984 BY a.woehrle"
100 PRINT@262,"                v.markmiller"
110 FOR R=1 TO 2000:NEXT R
120 CLEAR500
130 GOSUB2490
140 DIM KR(100):DIM KP(100)
150 FOR N=1 TO 900:NEXT
160 CLS:PRINT@33,"bitte waehlen sie die"
170 PRINT@65,"diagrammart"
180 PRINT@131,"SAEULENDIAGRAMM 1":PRINT@195,"BALKENDIAGRAMM  2":PRINT@259,"KREIS
DIAGRAMM   3":PRINT@323,"VERLAUFSKURVE    4":PRINT@387,"DIAGRAMM ABSPEICHERN    5"
:PRINT@451,"DIAGRAMM EINLESEN   6";
```

```
190 IF PEEK (339)=254 THEN GOTO 260
200 IF PEEK(340)=254 THEN GOTO 1680
210 IF PEEK(341)=254 THEN GOTO 1980
220 IF PEEK (342)=254 THEN GOTO 1280
230 IF PEEK(343)=254 THEN GOTO 2870
240 IF  PEEK(344)=254 THEN GOTO 3020
250 GOTO 190
260 REM saulendiagramm
270 GOSUB2330
280 MA=0
290 MI=0
300 FORI=1TO A
310 IF KR(I)>MA THEN MA=KR(I)
320 IFKR(I)<MI THEN MI=KR(I)
330 NEXT
340 XA=240/A
350 IF MA>0 THEN Y=(170-XA/4)/MA
360 IF MI<=0 THEN Y=(170-XA/4)/(MA+ABS(MI))
370 FH=180
380 IF MI<0 THEN FH=INT(180+MI*Y)
390 PMODE4,1 :SCREEN1,1:PCLS
400 LINE(7,FH)-(252,FH),PSET
410 GR=0:C=5:WA=1:SE=FH+2:K$="0":GOSUB2790
420 FOR I=1 TO A
430 B=I-1
440 X=INT(B*XA+XA/5)+5:Y1=INT(KR(I)*Y):XB=INT(XA/2):XC=INT(XA/4):XD=INT(XC/1.4)
450 DRAW"BM"+STR$(X)+","+STR$(FH)
460 IF KR(I)>=0 THEN DRAW"R"+STR$(XB)+"U"+STR$(Y1)+"L"+STR$(XB)+"ND"+STR$(Y1)+"M
+"+STR$(XC)+",-"+STR$(XD)+"R"+STR$(XB)
470 IF KR(I)<0 THEN DRAW"R"+STR$(XB)+"D"+STR$(ABS(Y1))+"L"+STR$(XB)+"U"+STR$(ABS
(Y1))+"M+"+STR$(XC)+",-"+STR$(XD)+"R"+STR$(XB)
480 IF KR(I)>=0 THEN DRAW"NM-"+STR$(XC)+",+"+STR$(XD)+"D"+STR$(Y1)+"M-"+STR$(XC)
+",+"+STR$(XD)
490 IF KR(I)<0 THEN DRAW"NM-"+STR$(XC)+",+"+STR$(XD)+"D"+STR$(ABS(Y1))+"M-"+STR$
(XC)+",+"+STR$(XD)
500 WA=X:SE=189:GR=0:C=5:K$=STR$(KR(I))
510 GOSUB2780
520 NEXTI
530 FOR Z=1 TO5000:NEXTZ
540 CLS:PRINT"WOLLEN SIE EINE HARDCOPY? (J/N)"
550 INPUT A$
560 IF A$="J" THEN GOTO 640
570 IF A$="N" THEN GOTO 590
580 GOTO 550
590 CLS:PRINT"WOLLEN SIE DIE GRAFIK WIEDER    SEHEN? (J/N)"
600 INPUT A$
610 IF A$="J" THEN GOTO 1920
620 IF A$="N" THEN GOTO 160
630 CLS
640 REM hardcopy"
650 DEF USR0=32500
660 PRINT#-2,CHR$(27);"A";CHR$(8);
670 FOR I=32500 TO 32593
680 READ INH:POKE I,INH
690 NEXTI
700 IF PEEK(1545)<>255 AND PEEK(1545)<>0 THEN POKE32505,&H0C:POKE32563,&H24
710 GOSUB 1210
720 X=USR0(0)
730 PRINT#-2,STRING$(5,CHR$(13))
740 CLS:PRINT@200,"ENDE DER AUSGABE"
750 GOTO160
760 REM**** MASCHINENCODE****
770 DATA &H34,&H7E
780 DATA &H32,&H75
790 DATA &H8E,&H06,&H00
800 DATA &H8D,&H3E
810 DATA &H1F,&H12
820 DATA &H31,&HA8,&H20
830 DATA &H10,&HAF,&H69
840 DATA &H1F,&H12
850 DATA &HC6,&H07
860 DATA &HE7,&H68
870 DATA &HA6,&HA4
880 DATA &HA7,&HE5
890 DATA &H31,&HA8,&H20
```

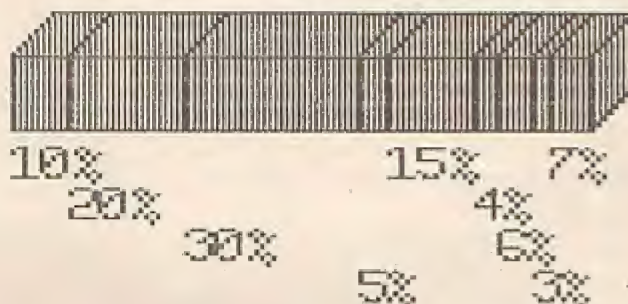



```
900 DATA &H5A
910 DATA &H2A,&HF6
920 DATA &HC6,&H07
930 DATA &H69,&HE5
940 DATA &H49
950 DATA &H5A
960 DATA &H2A,&HFA
970 DATA &H34,&H30
980 DATA &HBD,&H80,&H0F
990 DATA &H35,&H30
1000 DATA &H6A,&H68
1010 DATA &H2A,&HED
1020 DATA &H30,&H01
1030 DATA &HAC,&H69
1040 DATA &H26,&HD7
1050 DATA &H30,&H89,&H00,&HE0
1060 DATA &H8C,&H1E,&H00
1070 DATA &H26,&HC4
1080 DATA &H32,&H6B
1090 DATA &H35,&HFE
1100 DATA &H33,&H8C,&H0F
1110 DATA &HC6,&H05
1120 DATA &HA6,&HC0
1130 DATA &H34,&H54
1140 DATA &HBD,&H80,&H0F
1150 DATA &H35,&H54
1160 DATA &H5A
1170 DATA &H26,&HF4
1180 DATA &H39
1190 DATA &H0A,&H1B,&H4B,&H00
1200 DATA &H01
```

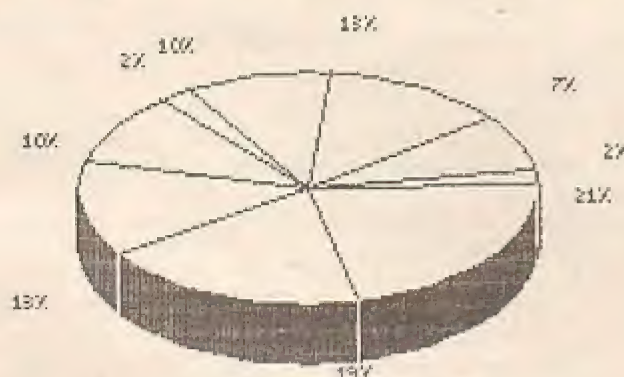



```
1210 CLS:PRINT@ 200,"DAS FOLGENDE BILD ";
1220 PRINT @264,"WIRD AUSGEGEBEN";
1230 FOR J=1 TO 500:NEXT J
1240 PMODE 4,1:SCREEN1,0
1250 FOR J=1 TO 1000:NEXT J
1260 RETURN
1270 REM verlaufskurve
1280 GOSUB2330
1290 MI=0
1300 MA=0:FOR N=1 TO A
1310 IF KR(N)>MA THEN MA=KR(N)
1320 IF KR(N)<MI THEN MI=KR(N)
1330 NEXT
1340 IF MA<>0 THEN Y=160/MA
1350 IF MI<0 THEN Y=160/(MA+ABS(MI))
1360 FH=180
1370 IF MI<0 THEN FH=INT(180+MI*Y)
1380 PMODE 4,1:SCREEN1,1:PCLS
1390 LINE(20,FH)-(250,FH),PSET
1400 LINE(20,180)-(20,10),PSET
1410 LINE(20,180)-(250,180),PSET
1420 T=240/A
1430 X=20
1440 IF MI>=0 THEN FOR N=0 TO MA+MA/20 STEP MA/10 ELSE FOR N=MI TO MA STEP (MA+A
BS(MI))/10
1450 LINE(18,FH-N*Y)-(22,FH-N*Y),PSET
1460 G$=STR$(INT(N*10)/10)
1470 WA=2:SE=FH-N*Y:GR=0:C=5:K$=G$:GOSUB2780
1480 IF ABS(N)>=10 THEN G$=STR$(INT(N))
1490 IF ABS(N)<0 THEN G$=STR$((INT(N*100)/100)
1500 NEXT
1510 FOR Z=1 TO A-1
1520 LET X=X+T
1530 LINE (X-T,FH-KR(Z)*Y)-(X,FH-KR(Z+1)*Y),PSET
1540 WA=INT(X-T):SE=189:K$=STR$(Z):K$=RIGHT$(K$,LEN(K$)-1):GOSUB2780
1550 LINE(X-T,182)-(X-T,178),PSET
1560 NEXTZ
1570 WA=INT(X):SE=189:K$=STR$(A):K$=RIGHT$(K$,LEN(K$)-1):GOSUB2780
1580 LINE(X,182)-(X,178),PSET
1590 FOR R=1 TO 5000:NEXT R
1600 CLS:PRINT"WOLLEN SIE EINE HARDCOPY? (J/N)"
1610 INPUT A$
1620 IF A$="J" THEN GOTO 640
1630 IF A$="N" THEN GOTO 1640
1640 CLS:PRINT"WOLLEN SIE DIE GRAFIK WIEDER    SEHEN? (J/N)"
1650 INPUT A$
```

```
1660 IF A$="J" THEN GOTO 1920
1670 IF A$="N" THEN GOTO 160
1680 REM balkendiagramm
1690 GOSUB2330
1700 PCLS:PMODE3,1:SCREEN1,0
1710 X=14:Y=105:D=2
1720 K=16
1730 FOR N=1 TO A
1740 DRAW"C4;BM"+STR$(INT(X))+","+STR$(INT(Y))+";"
1750 DRAW"R"+STR$(INT(D*KP(N)))+"U25E15L"+STR$(INT(D*KP(N)))+"G15ND25R"+STR$(INT
(D*KP(N)))
1760 GR=7:C=0:WA=INT(X):SE=INT(Y)+K:K$=STR$(INT(KP(N)))+"%":K$=RIGHT$(K$,LEN(K$)
-1):GOSUB2780:K=K+14:IF K>=72 THEN K=16
1770 IF INT(KP(N)*2)>=4 THEN PAINT(X+3,Y-2),2,4
1780 IF INT(KP(N)*D)>=4 THEN PAINT(X+3,Y-26),2,4
1790 X=X+INT(D*KP(N))
1800 NEXT
1810 DRAW"BM"+STR$(X)+","+STR$(Y)+"E15U25"
1820 PAINT(X+3,Y-5),2,4
1830 FOR N=1 TO 5000:NEXT
1840 CLS:PRINT"WOLLEN SIE EINE HARDCOPY? (J/N)"
1850 INPUT A$
1860 IF A$="J" THEN GOTO 640
1870 CLS:PRINT"WOLLEN SIE DIE GRAFIK WIEDER    SEHEN? (J/N)"
1880 INPUTA$
1890 IF A$="J" THEN GOTO 1920
1900 IF A$="N" THEN GOTO 160
1910 STOP
1920 CLS :PRINT "MIT m KOMMEN SIE WIEDER IN DAS  MENUE"
1930 FOR R=1 TO 1000:NEXT
1940 SCREEN1,0
1950 A$=INKEY$
1960 IF A$="M" THEN GOTO 160
1970 GOTO 1940
1980 REM *** kreis ***
1990 GOSUB2330
2000 PI=3.141592653627
2010 R=80
2020 PMODE4,1:SCREEN1,1
2030 PCLS
2040 MX=127:MY=95
2050 CIRCLE(MX,MY),R,,.5
2060 CIRCLE(MX,MY+20),R,,.5,0,0.5
2070 LINE(47,95)-(47,115),PSET
2080 LINE(207,95)-(207,115),PSET
2090 PAINT(127,145)
2100 P=0
2110 X1=207:Y1=95
2120 FOR N=1 TO A
2130 P=P+KP(N)
2140 WI=3.6*P
2150 R=79
2160 IF WI>180 GOTO 2250
2170 BO=(PI*WI)/180
2180 X=R*COS(BO)
2190 X1=R*2*COS(BO)
2200 Y=R*SIN(BO)
2210 Y1=R*2*COS(BO)
2220 IF WI<=180 THEN LINE(MX,MY)-(INT(MX+X),INT(MY-Y/2)),PSET
2230 GR=0:C=5:WA=INT(MX+X*1.3+2):SE=INT(MY-Y/2*1.3-2):K$=STR$(INT(KP(N)))+"%":K$
=RIGHT$(K$,LEN(K$)-1):GOSUB2780
2240 GOTO2310
2250 BO=(PI*WI)/180
2260 X=R*COS(BO)
2270 Y=R*SIN(BO)
2280 LINE(MX,MY)-(INT(MX+X),INT(MY-Y/2)),PSET
2290 LINE(INT(MX+X),INT(MY-Y/2))-(INT(MX+X),INT(MY-Y/2)+21),PRESET
2300 GR=0:C=5:K$=STR$(INT(KP(N)))+"%":K$=RIGHT$(K$,LEN(K$)-1):WA=INT(MX+X*1.35-1
-4*LEN(K$)):SE=INT(MY-Y/2*1.55+6):GOSUB2780
2310 NEXT
2320 GOTO530
2330 REM *** eingabe ***
2340 CLS:IF GE<>0 THEN PRINT"WOLLEN SIE ALTE WERTE BENUETZEN? (J/N)";
2350 IF GE<>0 THEN INPUT I$:IF I$="J" GOTO 2470
2360 CLS:PRINT"WOLLEN SIE DIE WERTE IN PROZENT ODER ALS ABSOLUTZAHLEN EINGEBEN (
p/a)                            BEI SAEULENDIAGRAMMEN UND DER   VERLAUFSGRAFIK SI
ND NUR ABSOLUT-WERTE MOEGLICH":INPUTI$
```



```
2370 CLS:A=1:PRINT"GEBEN SIE BITTE DIE WERTE EIN.  AM ENDE DANN BITTE x."
2380 INPUT C$:IF C$="X" THEN GOTO 2400 ELSE IF I$="P" THEN KP(A)=VAL(C$) ELSE KR
(A)=VAL(C$)
2390 A=A+1:GOTO2380
2400 A=A-1:IF I$="P" GOTO 2470
2410 GE=0:FOR N=1 TO A
2420 GE=GE+ABS(KR(N))
2430 NEXT
2440 FOR N=1 TO A
2450 KP(N)=ABS(KR(N))/GE*100
2460 NEXT
2470 GE=1
2480 RETURN
2490 REM *** zeichensaetze ***
2500 DIM A$(57),B$(57)
2510 RESTORE
2520 A$(46)="BM+1,+0;M+0,+0"
2530 A$(49)="R2L1U4G1"
2540 A$(50)="NR2U1E1E1H1L1"
2550 A$(51)="R1E1H1L1R1E1H1L1"
2560 A$(52)="BM+1,+0;U4G2R3"
2570 A$(53)="R1E1H1L1U2R2"
2580 A$(54)="BM+0,-1;F1R1E1H1L1G1U2E1R1"
2590 A$(55)="U1E3L3"
2600 A$(56)="BM+0,-1;F1R1E1H1L1G1E1H1E1R1F1G1"
2610 A$(57)="R1E1U2H1L1G1F1R1E1"
2620 A$(37)="BM+0,-4;M+0,+0;BM+0,+4;E1U1R1U1E1;BM+0,+4;M+0,+0"
2630 A$(48)="BM+0,-1;F1R1E1U2H1L1G1D1"
2640 A$(45)="BM+0,-2;R1"
2650 B$(46)="BM+1,+0;R1"
2660 B$(49)="R4L2U5L1G1"
2670 B$(50)="NR4U1E1R2E1U1H1L2G1"
2680 B$(51)="BM+0,-1;F1R2E1U1H1L1R1E1H1L2G1"
2690 B$(52)="BM+3,+0;U5G3R4"
2700 B$(53)="BM+0,-1;F1R2E1U1H1L3U2R4"
2710 B$(54)="BM+0,-1;F1R2E1H1L2G1U3E1R2F1"
2720 B$(55)="U1E4L4D1"
2730 B$(56)="BM+0,-1;F1R2E1U1H1L2G1D1U1E1H1E1R2F1G1"
2740 B$(57)="BM+0,-1;F1R2E1U3H1L2G1F1R2E1"
2750 B$(37)="E4BM-4,-0;EFGHBM+4,+4;GHEF"
2760 B$(48)="BM+0,-1;F1R2E1U3H1L2G1D3E3"
2770 RETURN
2780 REM *** ausgabe ***
2790 FOR U=1 TO LEN(K$)
2800 Q=ASC(MID$(K$,U,1))
2810 IF GR=0 THEN DRAW"S4BM"+STR$(WA)+","+STR$(INT(SE))+";"+"C"+STR$(C)+A$(Q) EL
SE DRAW"S8BM"+STR$(WA)+","+STR$(SE)+";"+"C"+STR$(C)+B$(Q)+"S4"
2820 WA=WA+4+GR:IF WA>255 THEN SE=SE+8:WA=0
2830 IF Q=46 THEN WA=WA-1
2840 NEXT U
2850 RETURN
2860 REM abspeichern der grafik"
2870 CLS
2880 PRINT"BENUTZEN SIE CASETTE ODER       DISKETTE? (C/D)"
2890 INPUT A$
2900 IF A$="C" THEN GOTO 2920
2910 IF A$="D" THEN GOTO 2980
2920 CLS:PRINT"NAME DES DIAGRAMMES BITTE?"
2930 INPUT A$
2940 CLS:PRINT"BITTE MACHEN SIE IHREN RECORDER BEREIT UND DRUECKEN DANN s.     N
ACH DEM ABSPEICHERN KOMMEN SIE MIT m WIEDER IN DAS MENUE."
2950 IF INKEY$="S" THEN CSAVEM"A$",1536,7679,1536
2960 IF INKEY$="M" THEN GOTO 160
2970 GOTO 2950
2980 CLS:PRINT"BITTE LEGEN SIE EINE DISKETTE   EIN UND DRUECKEN DANN s.        N
ACH DEM ABSPEICHERN KOMMEN SIE MIT m WIEDER IN DAS MENUE."
2990 IF INKEY$="S" THEN !"DIAGRAMM",3072,7680,3072
3000 IF INKEY$="M" THEN GOTO 160
3010 GOTO 2990
3020 REM ***einlesen von grafik**
3030 CLS:PRINT"VON DISK ODER CASSETTE? (D/C)"
3040 INPUT I$
3050 PMODE4,1:IF I$="D" THEN LOAD"DIAGRAMM.BIN" ELSE CLS:INPUT"NAME DER GRAFIK?"
;I$:CLOADM I$
3060 SCREEN1,1
3070 GOTO530
```



SNAKE

für Sharp MZ – 700

Sinn des Spieles ist es, eine Schlange in kürzester Zeit möglichst viele Äpfel und Kirschen fressen zu lassen. Dabei müssen Sie darauf achten, daß die Schlange nicht ein Hindernis, den Spielfeldrand oder sich selbst berührt. Geschieht dies trotzdem, verliert sie sofort ein Leben.
Sobald sich die Schlange in Bewegung setzt, läuft ein Zeitbalken ab. Gelingt es ihr, während des Ablaufs eine Frucht zu fressen, wird der Balken erneuert, wenn nicht, erfolgt Punktabzug.
Das Programm verfügt über drei Schwierigkeitsgrade und die Punkte- und max. Punkteanzeige (Highscore).

Tips zur Eingabe des Programms:

In manchen Zeilen steht eine PRINT-Anweisung mit Anführungszeichen, in denen schwarz unterstrichene Ziffern und Buchstaben gedruckt sind. Dies sind zweistellige Hexadezimal-Zahlen, die nebeneinander stehen. Ihr dezimaler Wert entspricht einem Zeichen aus dem ASCII-Code, welches der Plotter leider nicht ausgibt.
Die nun folgende Tabelle zeigt die Hexadezimal-Zahlen und das dazugehörige Zeichen.

Hex.	= Dez.	= ASCII Zeichen
63	99	
82	130	
87	135	
91	145	
C1	193	
C2	194	
C3	195	
C4	196	
C5	197	
C7	199	
C8	200	
C9	201	
CD	205	
CE	206	
DO	208	

Hex.	= Dez.	= ASCII Zeichen
D5	213	
D8	216	
DA	218	
DD	221	
E9	233	
EC	236	
ED	237	
EE	238	
EF	239	
FO	240	
F5	245	
FE	254	

Listing nächste Seite

```
10 REM ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
20 REM ~  (C)      SNAKE     (02.84) ~
30 REM ~                             ~
40 REM ~      Copyright by           ~
50 REM ~                             ~
60 REM ~      Uwe Nöther             ~
70 REM ~      Seelandstr. 4          ~
80 REM ~      8521 Oberreichenbach   ~
90 REM ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
100 COLOR ,,6,0
110 X=4: CONSOLE: TEMPO 7
120 MO=53248: FA=2048
130 PRINT"▩":
                          REM GRAFIK
140 S$=" EEC3C3C3ED▩▩▩▩▩▩▩EEEEC3C3C3EDC5▩▩▩▩▩▩▩▩C7C7▩▩▩C2C7▩▩EDC3C3C3C3ED▩▩▩▩▩▩C3C3C3F0C7▩▩▩C7C7▩▩▩▩▩▩▩C7EDC4C4C4EEEE▩▩▩▩▩▩▩EDC4C4C4EE"
150 S1$="      ▩▩▩▩▩▩▩      ▩▩▩▩▩▩▩▩ ▩▩▩ ▩▩    ▩▩▩▩▩▩    ▩▩▩ ▩▩▩▩▩▩▩▩     ▩▩▩▩▩▩▩     "
160 FOR A=15 TO 1 STEP-1: MUSIC"-C0"
170 CURSOR A+2,A: PRINT S1$
180 CURSOR A+1,A-1: PRINT S$
190 NEXT A
200 N$="C7F0  EFC5▩▩▩▩▩▩▩▩C7 ED  C5C5▩▩▩▩▩▩▩▩C7C7EDED C5C5▩▩▩▩▩▩▩▩C7C7 EDEDC5C5▩▩▩▩▩▩▩▩C7C7  ED C5▩▩▩▩▩▩▩▩C7DA  ECC5"
210 N1$="       ▩▩▩▩▩▩▩▩      ▩▩▩▩▩▩▩▩      ▩▩▩▩▩▩▩▩      ▩▩▩▩▩▩▩▩      ▩▩▩▩▩▩▩▩      "
220 FOR A=15 TO 1 STEP-1: MUSIC"-E0"
230 CURSOR A+10,A+3: PRINT N1$
240 CURSOR A+9,A+2: PRINT N$
250 NEXT A
260 E$="EFC3C3C3C5▩▩▩▩▩▩C5FDC3C3▩▩▩▩▩C5CDE0CE▩▩▩▩▩C5D0E0DD▩▩▩▩▩C5FDC4C4▩▩▩▩▩FCC4C4C4C5"
270 E1$="      ▩▩▩▩▩▩    ▩▩▩▩▩    ▩▩▩▩▩    ▩▩▩▩▩    ▩▩▩▩▩    "
280 FOR A=0 TO 14: MUSIC"-G0"
290 CURSOR A+17,18-A: PRINT E1$
300 CURSOR A+18,17-A: PRINT E$
310 NEXT A
320 K$="C4   C4▩▩▩▩▩▩C5C5 EEEE▩▩▩▩▩▩C5C5EEEE▩▩▩▩▩C5EEEE▩▩▩▩C5EDED▩▩▩▩C5C5EDED▩▩▩▩C5C5 EDED▩▩▩▩▩▩C3   C3"
330 K1$="    ▩▩▩▩▩▩    ▩▩▩▩▩▩   ▩▩▩▩▩   ▩▩▩▩   ▩▩▩▩   ▩▩▩▩▩   ▩▩▩▩▩▩   "
340 FOR A=0 TO 13: MUSIC"-B0"
350 CURSOR A+11,16-A: PRINT K1$
360 CURSOR A+12,15-A: PRINT K$
370 NEXT A
380 A$=" EEC3ED▩▩▩▩▩▩ EEEEC3EDED▩▩▩▩▩▩▩C7 ECC4DAC7▩▩▩▩▩▩▩C7 C4C4C4C7▩▩▩▩▩▩▩C7C7 C2C7▩▩▩▩▩▩▩C7DA  C7DA"
390 A1$="    ▩▩▩▩▩▩    ▩▩▩▩▩▩▩    ▩▩▩▩▩▩▩    ▩▩▩▩▩▩▩    ▩▩▩▩▩▩▩    "
400 FOR A=1 TO 15: MUSIC"D0"
410 CURSOR 17,19-A: PRINT A1$
420 CURSOR 17,18-A: PRINT A$
430 NEXT A
440 MUSIC"-C0-D-E-F-G-A-BCDEFGAB+C+D+E+F+G+A+B"
450 SS$=" F5C8C8C8E9▩▩▩▩▩▩▩F5C9C3C3C3FEC5▩▩▩▩▩▩▩▩C7C8▩▩▩C2C8▩▩FEC8C8C8E9▩▩▩▩▩▩C3C3C3F0C8▩▩▩C7C8▩▩▩▩▩▩▩C7E9C4C4C4F5C9▩▩▩▩▩▩▩FEC8C8C9"
460 NN$="C7C8C5  C8C5▩▩▩▩▩▩▩▩C7C8E9  C8C5▩▩▩▩▩▩▩▩C7C8FEE9 C8C5▩▩▩▩▩▩▩▩C7C8 FEE9C8C5▩▩▩▩▩▩▩▩C7C8  FEC8C5▩▩▩▩▩▩▩▩C7C8   C8C5"
470 EE$="C8C8C8C8C5▩▩▩▩▩▩C8   ▩▩▩▩▩C8C2C2 ▩▩▩▩▩C8D8C8 ▩▩▩▩▩C8   ▩▩▩▩▩C8C8C8C8C5"
480 KK$="C4   C4▩▩▩▩▩▩C8  F5C9▩▩▩▩▩▩C8 F5C9▩▩▩▩▩C8F5C9▩▩▩▩C8FEE9▩▩▩▩C8 FEE9▩▩▩▩▩C8  FEE9▩▩▩▩▩▩C3   C3"
490 AA$="  F5C8E9▩▩▩▩▩▩ F5C9C3FEE9▩▩▩▩▩▩▩C7C8ECC4DAC9▩▩▩▩▩▩▩C7C8C8C8C8C8▩▩▩▩▩▩▩C2C8  C2C8▩▩▩▩▩▩▩C2C8  C2C8"
500 GOTO 630
510 CURSOR 2,0: PRINT [X,] S$
520 CURSOR 10,3: PRINT [X,] N$
530 CURSOR 17,3: PRINT [X,] A$
540 CURSOR 25,2: PRINT [X,] K$
550 CURSOR 32,3: PRINT [X,] E$
560 CURSOR 2,0: PRINT [X,] SS$
570 CURSOR 10,3: PRINT [X,] NN$
580 CURSOR 17,3: PRINT [X,] AA$
590 CURSOR 25,2: PRINT [X,] KK$
600 CURSOR 32,3: PRINT [X,] EE$
610 X=INT(RND(1)*6)+1
620 RETURN
630 FOR A=1 TO 20: GOSUB 510: NEXT A
640 CURSOR 16,20: PRINT"Beschreibung Ja/Nein"
650 GET Z$: IF Z$="" THEN GOSUB 510: GOTO 650
660 IF Z$="N" THEN 1150
670 IF Z$<>"J"THEN 650
680 CONSOLE 11,14,3,33:
                          REM Beschreibung
690 PRINT"▩Bei diesem Spiel müssen Sie ver-▩"
700 PRINT"suchen möglichst viele Äpfel und▩"
710 PRINT"Kirschen zu fressen.▩"
720 PRINT"Berühren Sie den Spielfeldrand.▩"
730 PRINT"sich selber oder ein Hindernis.▩"
740 PRINT"verlieren Sie ein Leben !▩▩"
750 PRINT TAB(20);"Taste drücken !▩"
760 CONSOLE: MUSIC"+B"
770 GET Z$: IF Z$="" THEN GOSUB 510: GOTO 770
780 CONSOLE 11,14,3,33
790 PRINT"▩Ein Zeitbalken läuft auch mit.▩"
800 PRINT"Er wird kleiner sobald Sie die▩"
810 PRINT"Schlange mit den Cursor-Tasten▩"
820 PRINT"bewegen.▩▩▩▩"
830 PRINT TAB(20);"Taste drücken !▩"
840 CONSOLE: MUSIC"+B"
850 GET Z$: IF Z$="" THEN GOSUB 510: GOTO 850
860 CONSOLE 11,14,3,33
870 PRINT"▩Fressen Sie eine Frucht, bevor▩"
880 PRINT"der Zeitbalken abgelaufen ist,▩"
890 PRINT"beginnt er von Vorne.▩"
900 PRINT"Schaffen Sie es nicht, gibt`s▩"
910 PRINT"Strafpunkte !▩▩"
920 PRINT TAB(20);"Taste drücken !▩"
930 CONSOLE: MUSIC"+B"
940 GET Z$: IF Z$="" THEN GOSUB 510: GOTO 940
950 CONSOLE 11,14,3,33
960 PRINT"▩        82 ...... 100 Punkte▩"
```

```
970 PRINT"         82 ...... 200 Punkte▯▯"
980 PRINT" Ist der Zeitbalken abgelaufe
n▯"
990 PRINT"          minus 100 Punkte ???▯▯
▯"
1000 PRINT TAB(20);"Taste drücken ?B"
1010 POKE MO+FA+450,192
1020 POKE MO+FA+530,176
1030 CONSOLE: MUSIC"+B"
1040 GET Z$: IF Z$="" THEN GOSUB 510: GO
TO 1040
1050 CONSOLE 11,14,3,33
1060 PRINT"▯Nach Spielende können Sie de
n▯"
1070 PRINT"Schwierigkeitsgrad durch drüc
ken▯"
1080 PRINT"der Taste 1, 2 oder 3 ändern.
▯"
1090 PRINT"Wollen Sie den Schwierigkeits
-▯"
1100 PRINT"grad so belassen, dann drücke
n ▯"
1110 PRINT"Sie eine andere Taste.▯"
1120 PRINT TAB(20);"Taste drücken ?B"
1130 CONSOLE: MUSIC"+B"
1140 GET Z$: IF Z$="" THEN GOSUB 510: GO
TO 1140
1150 CONSOLE 11,14,3,33
1160 PRINT"▯▯▯        Schwierigkeitsgrad
▯"
1170 PRINT"             1 / 2 / 3▯"
1180 PRINT"                  ?"
1190 CONSOLE: MUSIC"+B"
1200 GET Y: IF (Y=1)+(Y=2)+(Y=3) THEN 12
20
1210 GOSUB 510: GOTO 1200
1220 MUSIC"-D-E-F-E-D": CURSOR 20,22: PR
INT"Taste drücken ?"
1230 GET Z$: IF Z$="" THEN GOSUB 510: GO
TO 1230
1240 IF Y=1 THEN LA=8
1250 IF Y=2 THEN LA=6
1260 IF Y=3 THEN LA=4
1270 CLS:
                    REM ? Schwierigk.
1280 FOR B=1 TO 50
1290 X=INT(RND(1)*6+1)
1300 COLOR ,,X,: CURSOR 10,12
1310 PRINT [X,0];"SCHWIERIGKEITSGRAD ";Y
1320 FOR A=1 TO 20: NEXT A
1330 NEXT B
1340 MUSIC"C0DEFEFE"
1350 COLOR ,,6,0
1360 PRINT"▯ 636363  Punkte:      0  max.
Punkte: ";USING"#####";MP
1370 TEMPO 7
1380 DIM A(8),MD(20),H(1),F(1)
1390 FOR A=1 TO 8: A(A)=53247: NEXT A
1400 PL=1:
                    REM Spiel 1
1410 AD=53489: I=10: E=0: H(0)=157: H(1)
=158: F(1)=176: F(0)=192

1420 GOSUB 1940:
                    REM Rahmen
1430 ON PL GOSUB 2540,2620,2700:
                    REM Hindernisse
1440 IF PL=4 THEN  GOSUB 2720: GOTO 1430
1450 POKE AD,207:
                    REM Schlangenkopf
1460 GOSUB 2300:
                    REM Fruechte
```

```
1470 GET A$: IF A$="" THEN 1470
1480 POKE AD,0: GOTO 1500
1490 GET A$
1500 IF A$="▯" THEN 1580
1510 IF A$="▯" THEN 1660
1520 IF A$="▯" THEN 1740
1530 IF A$="▯" THEN 1820
1540 GOSUB 2390:
                    REM Fruechte CLR
1550 ON A GOTO 1580,1660,1740,1820,1410
1560 GOTO 1490
1570 REM ~~~~~~~~~~ ▯▯▯▯▯▯▯▯▯ ~~~~~~~~~~
~
1580 AD=AD-40
1590 IF PEEK(AD)<>0 GOSUB 2110
1600 IF A=2 THEN POKE AD,207: POKE AD+40
,187: GOTO 1630
1610 IF A=4 THEN POKE AD,207: POKE AD+40
,184: GOTO 1630
1620 POKE AD,207: IF I=0 THEN POKE AD+40
,238
1630 A=1: I=0
1640 GOTO 1890
1650 REM ~~~~~~~~~~ ▯▯▯▯▯▯▯▯▯ ~~~~~~~~~~
~
1660 AD=AD+1
1670 IF PEEK(AD)<>0 GOSUB 2110
1680 IF A=3 THEN POKE AD,207: POKE AD-1,
184: GOTO 1710
1690 IF A=1 THEN POKE AD,207: POKE AD-1,
156: GOTO 1710
1700 POKE AD,207: IF I=0 THEN POKE AD-1,
227
1710 A=2: I=0
1720 GOTO 1890
1730 REM ~~~~~~~~~~ ▯▯▯▯▯▯▯▯▯ ~~~~~~~~~~
~
1740 AD=AD+40
1750 IF PEEK(AD)<>0 GOSUB 2110
1760 IF A=2 THEN POKE AD,207: POKE AD-40
,159: GOTO 1790
1770 IF A=4 THEN POKE AD,207: POKE AD-40
,156: GOTO 1790
1780 POKE AD,207: IF I=0 THEN POKE AD-40
,238
1790 A=3: I=0
1800 GOTO 1890
1810 REM ~~~~~~~~~~ ▯▯▯▯▯▯▯▯▯ ~~~~~~~~~~
~
1820 AD=AD-1
1830 IF PEEK(AD)<>0 GOSUB 2110
1840 IF A=3 THEN POKE AD,207: POKE AD+1,
187: GOTO 1870
1850 IF A=1 THEN POKE AD,207: POKE AD+1,
159: GOTO 1870
1860 POKE AD,207: POKE AD+1,227
1870 A=4

1880 REM ~~~~~~~~~~~ SCHWANZ ~~~~~~~~~~
1890 POKE A(LA),0
1900 FOR B=LA TO 2 STEP-1: A(B)=A(B-1):
NEXT
1910 A(1)=AD
1920 GOTO 1560
1930 REM ~~~~~~~~~~~~ RAHMEN ~~~~~~~~~~~
1940 IF PL>1 THEN RETURN
1950 PRINT"B"
1960 FOR A=0 TO 39: IF PEEK(MO+A)=0 THEN
 POKE MO+A,240
1970 NEXT A
1980 PRINT [2,] "C8" TAB(38) "C8"
1990 CURSOR 0,24
2000 PRINT [2,] "C8" TAB(38) "C8B"
```

```
2010 FOR A=1 TO 37:PRINT [2,] TAB(A) "D8
";: NEXT A: PRINT
2020 FOR A=1 TO 22: PRINT [2,] "D5" TAB(
38) "C1": NEXT
2030 FOR A=1 TO 37:PRINT [2,] TAB(A) "C2
";: NEXT A
2040 PRINT"B"
2050 FOR A=40 TO 1000 STEP 40: POKE MO+A
-1,240: NEXT A
2060 CURSOR 37,8
2070 PRINT [2,];"F5C8@@@D5F@@@D5@@D5@@D5
@@D5@@D5@@D5@@D5F@@@FEC8"
2080 GOSUB 2910:
                         REM Zeitbalken
2090 RETURN
2100 REM ~~~~~~~~ PUNKTE SAMMELN ~~~~~~~
2110 IF PEEK(AD)=157 THEN PU=PU+100: POK
E AD,0: POKE AD+FA,96: GOTO 2240
2130 GOSUB 2980:
                         REM Spiel verlo.
2140 REM ~~~~~~~~~ Wurm CLR ~~~~~~~~~~~~
2150 MUSIC"R0BAGFEDC-B-A-G-F-E-D-C"
2160 FOR I=1 TO 7: X=INT(RND(1)*7)
2170 FOR Z=8 TO 1 STEP-1: POKE A(Z)+FA,1
5+16*X: NEXT Z: NEXT I
2180 FOR Z=8 TO 1 STEP-1: POKE A(Z)+FA,9
6
2190 POKE A(Z),0: A(Z)=MO: NEXT Z
2200 AD=53489: I=10: A=0: POKE AD,207: F
R=FR+1: IF FR=3 THEN 2750
2210 GOSUB 2910:
                         REM Zeitbalken
2220 GOTO 1470:
                         REM Spielanfang
2230 REM ~~~~~~~~ Punktanzeige ~~~~~~~~~
2240 MUSIC"+B0"
2250 PRINT"B";TAB(14);USING"#####";PU: I
F PU<0 THEN 2940:     REM Spiel verlo.
2260 IF MP<PU THEN MP=PU: PRINT"@";TAB(3
3);USING"#####";MP
2270 E=E-1: GOSUB 2910:
                         REM Zeitbalken
2280 RETURN
2290 REM ~~~~~~~~~ Hindernisse ~~~~~~~~~
2300 XY=INT(RND(1)*878)+80: H=INT(RND(1)
*2)
2310 IF PEEK(MO+XY)<>0 THEN 2300
2320 MO(E)=MO+XY
2330 POKE MO(E),H(H): POKE MO(E)+FA,F(H)

2340 E=E+1: IF E=20 THEN E=19: RETURN
2350 GOTO 2300
2360 REM
2370 REM ~~~~~~~~~ Schlange CLR ~~~~~~~~
2380 REM
2390 IF E<>-1 THEN 2440
2400 MUSIC"+C2+F+D+C"
2410 FOR Z=1 TO 8: POKE A(Z),0: A(Z)=532
47: NEXT Z
2420 PL=PL+1: A=5: RETURN
2430 REM ~~~~~~~~~ Zeitbalken ~~~~~~~~~~
2440 T=T+1: IF T=3 THEN T=0: GOTO 2460
2450 RETURN
2460 IF PEEK(MO+438+ZB)=58 THEN POKE MO+
438+ZB,240: ZB=ZB+40: RETURN
2470 IF PEEK(MO+438+ZB)=67 THEN POKE MO+
438+ZB,58: RETURN
2480 IF PEEK(MO+438+ZB)=122 THEN POKE MO
+438+ZB,240: ZB=0: PU=PU-100
2490 MUSIC"-F0R1-F0R1-F0R1-F0R1-F0R9-F0R
1-F0R1-F0R1-F0R1-F0"
2500 E=E+1: GOSUB 2250:
                         REM Punktanzeige
2510 GOSUB 2910:
                         REM Zeitbalken
2520 RETURN
2530 REM ~~~~~~~~ HINDERNISSE ~~~~~~~~~~
2540 H1$="@@9191@@@@919191919@@@@@@91919
191919@@@@@@91919191@@@@919!"
2550 CURSOR 17,10: PRINT[1,5] H1$
2560 CURSOR 6,4: PRINT [6,0] H4$
2570 CURSOR 31,4: PRINT [6,0] H4$
2580 CURSOR 6,19: PRINT [6,0] H4$
2590 CURSOR 31,19: PRINT [6,0] H4$
2600 RETURN
2610 REM
2620 H2$="@91@@@91919@@@91"
2630 H3$="@@  @@@@     @@@@@@        @@@@@@
     @@@@  "
2640 CURSOR 17,10: PRINT [6,] H3$
2650 CURSOR 6,4: PRINT [1,5] H2$
2660 CURSOR 31,4: PRINT [1,5] H2$
2670 CURSOR 6,19: PRINT [1,5] H2$
2680 CURSOR 31,19: PRINT [1,5] H2$
2690 RETURN
2700 H4$="" : GOSUB 2540
2710 RETURN
2720 PL=1
2730 H4$="@ @@@   @@@ "
2740 RETURN
2750 REM ~~~~~~~~ Game over ~~~~~~~~~~~
2760 CURSOR 15,12: PRINT [7,]"GAME OVER"
2770 CURSOR 14,10: PRINT"C4C4C4C4C4C4C4C
4C4C4C4"
2780 FOR A=1 TO 3
2790 CURSOR 13,10+A: PRINT"C7@@@@@@@@@@@
C5"
2800 NEXT A
2810 CURSOR 14,14: PRINT"C3C3C3C3C3C3C3C
3C3C3C3"
2820 A$="          @@@@@@@@@@@@ GAME OV
ER @@@@@@@@@@@@
2830 CURSOR 14,11: X=INT(RND(1)*8)
2840 PRINT [0,X] A$
2850 GET B$: IF B$="" THEN 2830
2860 IF B$="1" THEN Y=1

2870 IF B$="2" THEN Y=2
2880 IF B$="3" THEN Y=3
2890 PU=0: FR=0: GOTO 1240
2900 REM ~~~~~~~~ Zeitbalken ~~~~~~~~~~
2910 CURSOR 38,10: PRINT [3,] "C2@@C8@@C
8@@C8@@C8@@D8": ZB=0
2920 RETURN
2930 REM ~~~~~~~~~~~~ PU=0 ~~~~~~~~~~~~~
2940 PU=0: E=E-1
2950 PRINT"B";TAB(14);USING"#####";PU
2960 GOTO 2130:
                         REM Spiel verlo.
2970 REM ~~~~~~~ Spiel verloren ~~~~~~~~
2980 FOR B=1 TO 15
2990 POKE MO+1+FR,202: IF PU=0 THEN POKE
 MO+18,240
3000 MUSIC"+A0"
3010 POKE MO+1+FR,240: IF PU=0 THEN POKE
 MO+18,32
3020 MUSIC"+B0"
3030 NEXT B
3040 POKE MO+FA+1+FR,96
3050 RETURN
```

*   *   *

Suchsel

für Apple II

Ein Spaß für alle Rätselfreunde

Das Programm SUCHSEL wurde auf einem APPLE II geschrieben und getestet. Da es aber ausschließlich in Basic geschrieben wurde, kann man es aber auch auf anderen Systemen zum Laufen bringen.

In *SUCHSEL* hat der Rätselrater die Aufgabe, in einem Buchstabenchaos Wörter zu finden, die entweder horizontal, vertikal, diagonal oder quer durchs Gemüsebeet verteilt sind. *SUCHSEL* ermöglicht nun den Heimcomputerfans selbst solche Rätsel zu erstellen. Dazu muß natürlich zunächst das Programm eingegeben werden. *Hierbei ist zu beachten:* Wenn das Programm nur an einem *APPLE* betrieben werden soll, an den mit Sicherheit ein Drucker angeschlossen ist, können die Zeilen 1010 und 1070 weggelassen werden. Andernfalls tippen Sie bitte die vorliegende Version ab und fügen noch die Zeile *„95 GOSUB 1010"* dazu. Besitzen Sie kein Diskettenlaufwerk, sollte man statt des Befehls *„PR ≠ 6"* jeweils die Folge *"HOME : END"* eintippen.
Nach fehlerfreier Eingabe speichere man zunächst das Programm und tippe Run. Es folgt entweder die Frage ob ein Drucker vorhanden ist, oder die Aufforderung, zwischen Laden eines Rätsels von Disk oder der Direkteingabe. Da beim erstenmal noch kein Rätsel gespeichert ist, wählt man Direkteingabe. Hierbei gibt man die Anzahl der Wörter ein, die in dem Feld aus 50 mal 50 (2500) Buchstaben verteilt werden sollen. Jetzt können die Wörter eingegeben werden. Maximallänge hierbei ist vierzig Zeichen. Aber es gibt ja auch kaum Wörter die diese Anzahl übersteigen, außer vielleicht

„Donaudampfschiffahrtskapitänswitwenrenteneinkommenssteuerrückzahlungsantragsformular"

oder sowas ähnliches. Weiterhin dürfen bei der Eingabe nur Großbuchstaben verwandt werden. Auch Leerstellen am Anfang oder Ende eines Wortes erweisen sich als äußerst ungünstig, denn der Computer nimmt an, sie gehören zum Wort. Ist im Rätselfeld jedes Wort klein geschrieben oder von Leerstellen eingerahmt, ist es allzu leicht auffindbar.
Die Frage des Computers, die irgendwann erscheint, ob die Wörter ausgedruckt werden sollen oder nicht, hat folgenden tieferen Sinn: Werden die Wörter mit ausgedruckt, so weiß man, wonach man suchen muß. Erfolgt kein Ausdruck, erschwert dies natürlich das Raten. Während der Computer die Wörter verteilt, erscheint hin und wieder ein Text auf dem Bildschirm, der Sie um etwas Geduld bittet. Sie können diese Zeit nutzen und überprüfen, ob der Drucker eingeschaltet (ON LINE) ist. Nach einiger Zeit des Wartens wird dann das Rätsel ausgedruckt. Anschließend folgt die Frage, ob das SUCHSEL auf Disk gespeichert werden soll, sofern man es nicht ohnehin von dort geladen hatte. In letzterem Fall unterbleibt diese Frage. Der Filename beim Laden und Speichern kann beliebig sein. Der *APPLE* verpaßt ihm automatisch das Präfix „RAETSEL". Zum Schluß folgt noch die Auswahl zwischen I (Index = Inhaltsverzeichnis), T (Terminate = Schluß) und R (Restart = Nochmal). Der Restart stößt dabei einfach auf einen RUN-Befehl, der Index führt einen warm reboot (PR ≠ 6) durch und Terminate beseitigt das ganze Programm mit NEW (daher auch die Aufforderung, es vorher zu speichern).
Wir wünschen Ihnen ein fehlerfreies abtippen und viel Glück beim Knacken des Rätsels.



Unser Suchrätsel

Folgende Wörter sind zu finden:
(horizontal, vertikal oder diagonal)

```
DATENBANK      HEIMCOMPUTER      SYSTEM

PROGRAMMBESCHREIBUNG     FARBMONITOR     LISTING     AUFMERKSAMKEIT
BUNDESPOST      HACKING      PROFESSIONAL      SPRACHSYNTHESIZER
PROGRAMMIERSPRACHE      SPRACHERKENNUNG      BETRIEBSSYSTEM
VIDEOSPIEL      ASSEMBLER      DATENERFASSUNG
DATEIVERWALTUNGSSYSTEM      TASTATURABFRAGE      DISKETTENBETRIEBSSYSTEM
DRUCKER     BILDSCHIRM     BASIC     PASCAL     FORTH     UNIX
DOS     EDV     CPU     TERMINAL
```

```
10  REM ***************
20  REM * SUCHRAETSEL *
30  REM ***************
40  REM * Alwin Ertl  *
50  REM * Mai 1984    *
60  REM ***************
70  HOME : INVERSE : PRINT "SUCHRAETSELGENERATOR": NORMAL
80 FL = 0: DIM F(50,50)
90 D$ =  CHR$ (4)
100  PRINT : PRINT "Von Disk laden oder Direkteingabe": INPUT "L oder D? "
     ;LD$
110  IF LD$ <  > "L" AND LD$ <  > "D" THEN 100
120  IF LD$ = "L" THEN 790
130  PRINT : PRINT : INPUT "Wieviele Woerter?";AZ
140  DIM W$(AZ)
150  FOR I = 1 TO AZ
160  PRINT "Wort ";I;":"
170  INPUT "";W$(I): IF  LEN (W$(I)) > 40 THEN  FLASH : PRINT "WORT ZU LAN
     G": NORMAL : PRINT "Neueingabe:": GOTO 170
180  NEXT I
190  PRINT : PRINT : INPUT "Woerter mit ausdrucken? ";AD$
200 AD$ =  LEFT$ (AD$,1): IF AD$ <  > "J" AND AD$ <  > "N" THEN 190
210  PRINT : PRINT : PRINT : PRINT  SPC( 15);: INVERSE : PRINT "BITTE WART
     EN": NORMAL
220  FOR I = 1 TO 50: FOR J = 1 TO 50:F(I,J) = 32: NEXT J,I
230  PRINT  SPC( 15);"NOCH'N MOMENT"
240  FOR I = 1 TO AZ
250 X =  INT (50 *  RND (1) + 1):Y =  INT (50 *  RND (1) + 1):C = 0
260 ZU =  RND (1)
270  IF ZU < .25 THEN XP = 1:YP = 0: GOTO 310
280  IF ZU < .5 THEN XP = 0:YP = 1: GOTO 310
290  IF ZU < .75 THEN XP = 1:YP = 1: GOTO 310
300 XP = 1:YP =  - 1
310 X1 = X:Y1 = Y
320  IF X1 > 50 OR X1 < 0 OR Y1 > 50 OR Y1 < 0 THEN 250
330 C = C + 1: IF F(X1,Y1) <  >  ASC ( MID$ (W$(I),C,1)) AND F(X1,Y1) <  >
     32 THEN 250
340 X1 = X1 + XP:Y1 = Y1 + YP
350  IF C <  LEN (W$(I)) THEN 320
360 F(X,Y) =  ASC ( LEFT$ (W$(I),1))
370  FOR J = 2 TO  LEN (W$(I))
380 X = X + XP:Y = Y + YP:F(X,Y) =  ASC ( MID$ (W$(I),J,1))
390  NEXT J,I
400  PRINT  SPC( 10);: FLASH : PRINT "GLEICH GEHT'S LOS": NORMAL
410  FOR I = 1 TO 50: FOR J = 1 TO 50
420  IF F(I,J) = 32 THEN F(I,J) =  INT (26 *  RND (1) + 65)
430  NEXT J,I
440  PRINT  SPC( 10);: INVERSE : PRINT "JETZT KOMMT'S !!!": NORMAL
450  PR# 1: PRINT  CHR$ (9);"80N"
460  PRINT "S u c h r a e t s e l": PRINT "=====================": PRINT
470  PRINT "vom Apple II e und ";: PRINT "_____________________": PRINT  SPC(
     20);"(Name des Autors)"
480  PRINT : PRINT
490  FOR I = 1 TO 50: FOR J = 1 TO 50
500  PRINT  CHR$ (F(I,J));
510  NEXT J: PRINT : NEXT I
520  PRINT : PRINT "Suchraetsel"
530  IF AD$ = "N" THEN 600
540  PRINT : PRINT : PRINT "Folgende Woerter sind zu finden": PRINT "(hori
     zontal, vertikal oder diagonal):": PRINT
550 PC = 0
560  FOR I = 1 TO AZ
570 PC = PC +  LEN (W$(I)) + 4:: IF PC > 50 THEN PC = 0: PRINT
580  PRINT W$(I);"    ";: NEXT I
590  PRINT  CHR$ (12)
600  PR# 0: IF FL THEN 960
610  HOME
620  PRINT : PRINT "Soll das Raetsel auf Disk": INPUT "gespeichert werden?
     ";SP$
630 SP$ =  LEFT$ (SP$,1): IF SP$ <  > "J" AND SP$ <  > "N" THEN 620
640  IF SP$ = "N" THEN 960
650  PRINT : PRINT "Unter welchem Filenamen?": INPUT "RAETSEL.";F$
660 F$ = "RAETSEL." + F$
```

```
670  PRINT : INPUT "Drive Nr. ";D
680  IF D < 1 OR D > 2 THEN 670
690  PRINT : PRINT "Wird gespeichert."
700  PRINT D$;"OPEN ";F$ + ",D"; STR$ (D)
710  PRINT D$;"WRITE ";F$
720  PRINT AZ
730  FOR I = 1 TO 50: FOR J = 1 TO 50
740  PRINT F(J,I)
750  NEXT J,I
760  FOR I = 1 TO AZ: PRINT W$(I): NEXT I
770  PRINT D$;"CLOSE ";F$
780  GOTO 960
790  PRINT : PRINT : PRINT "Welcher Filename?": INPUT "RAETSEL.";F$
800 F$ = "RAETSEL." + F$
810  PRINT : INPUT "Von Drive Nr. ";D
820  IF D < 1 OR D > 2 THEN 810
830  PRINT : PRINT "Wird geladen."
840  PRINT D$;"OPEN ";F$;",D"; STR$ (D)
850  PRINT D$;"READ ";F$
860  INPUT AZ
870  DIM W$(AZ)
880  FOR I = 1 TO 50: FOR J = 1 TO 50
890  INPUT F(I,J)
900  NEXT J,I
910  FOR I = 1 TO AZ: INPUT W$(I): NEXT I
920  PRINT D$;"CLOSE ";F$
930  PRINT : PRINT : INPUT "Woerter mit ausdrucken? ";AD$
940 AD$ =  LEFT$ (AD$,1): IF AD$ <  > "J" AND AD$ <  > "N" THEN 930
950 FL = 1: GOTO 450
960  HOME : PRINT "I)ndex": PRINT "T)erminate": PRINT "R)estart"
970  GET A$: IF A$ <  > "I" AND A$ <  > "T" AND A$ <  > "R" THEN 960
980  IF A$ = "I" THEN  PR# 6
990  IF A$ = "T" THEN  HOME : NEW
1000  CLEAR : RUN
1010  PRINT : PRINT "Ist ein Drucker angeschlossen?": INPUT "";DA$
1020 DA$ =  LEFT$ (DA$,1): IF DA$ <  > "J" AND DA$ <  > "N" THEN 1010
1030  IF DA$ = "N" THEN  PRINT : PRINT "Dann kannst Du dieses Programm nic
     ht": PRINT "benutzen. Eine Taste bringt Dich zum": PRINT "Inhaltsverz
     eichnis.": GET A$: PR# 6
1040  PRINT : PRINT "Ist der Drucker eingeschaltet?": INPUT "";DA$
1050 DA$ =  LEFT$ (DA$,1): IF DA$ <  > "J" AND DA$ <  > "N" THEN 1040
1060  IF DA$ = "N" THEN  PRINT : PRINT "Dann schalte ihn gefaelligst ein !
     ": PRINT "Anschliessend Taste druecken.": GET A$
1070  RETURN
```

ASC II-Dumper

für Apple II

Wir haben hier ein Utility-Programm für den Apple II und seine kompatiblen Nachbauten.

Es dient zur Ausgabe von Speicherauszügen im ASCII-Format, wenn man Texte von unbekannten Maschinen-Routinen sucht. Nach dem Start erscheint der Programmname und der Hinweis, daß man mit Hilfe des &-Befehls von Basic aus den ASCII-Dumper einsetzen kann. Vom Monitor aus wird mit CTRL-Y die Dump-Routine aufgerufen. Nach dem Aufruf muß man die Anfangsadresse eingeben, von der an der ASCII-Dump ausgegeben werden soll. Nun erscheinen die Inhalte der Speicherstellen und die Adresse des ersten Zeichens.
Durch drücken der Space-Taste wird die Ausgabe unterbrochen, bis die Taste erneut gedrückt wird. Bei Eingabe von ASCape wird die Routine neu gestartet und man kann eine neue Startadresse eingeben. Drückt man Return, wird der Programmablauf beendet. Mit & oder CTRL-Y ist ein erneuter Aufruf möglich.

Eingabe des Programms:

CALL -151 (RETURN)

8000: 20 99 80 20 C6 80 ...

Abspeichern auf Disketten:

BSAVE ASCII-Dumper, A$8000, L$16C

Listing: ASCII-Dumper

```
*
8000- 20 99 80 20 C6 80 A9 21
8008- A2 80 A0 4C 8C F5 03 8D
8010- F6 03 8E F7 03 8C F8 03
8018- 8D F9 03 8E FA 03 4C 03
8020- E0 20 99 80 A9 04 85 22
8028- 20 A4 80 20 58 FC 20 8E
8030- FD A6 E8 A5 E9 20 41 F9
8038- A2 02 20 4A F9 A9 AD 20
8040- ED FD A2 02 20 4A F9 A0
8048- 00 B1 E8 C9 8D D0 02 A9
8050- AE 20 ED FD C8 C0 10 D0
8058- F0 A5 E8 18 69 10 85 E8
8060- A5 E9 69 00 85 E9 A9 1E
8068- 20 A8 FC AD 00 C0 2C 10
8070- C0 10 BB C9 A0 F0 15 C9
8078- 1B F0 AD C9 8D D0 AF 20
8080- 58 FC A9 00 85 22 20 C6
8088- 80 4C 03 E0 AD 00 C0 2C
8090- 10 C0 C9 A0 D0 F6 4C 2E
8098- 80 20 58 FC A9 DC A0 80
80A0- 20 3A DB 60 20 58 FC 20
80A8- 22 FC A9 2D A0 81 20 3A
80B0- DB 20 6F FD E0 04 D0 EC
80B8- A0 00 20 A7 FF A5 3E 85
80C0- E8 A5 3F 85 E9 60 A9 0C
80C8- 85 25 20 22 FC A9 48 A0
80D0- 81 20 3A DB A9 14 85 25
80D8- 20 22 FC 60 C1 D3 C3 C9
80E0- C9 AD C4 F5 ED F0 E5 F2
80E8- A0 A0 A8 C3 A9 A0 B1 B9
80F0- B8 B4 A0 E2 F9 A0 CF AE
80F8- A0 D3 F4 E5 E9 EE ED E5
8100- E9 E5 F2 A0 BD BD BD BD
8108- BD BD BD BD BD BD BD BD
8110- BD BD BD BD BD BD BD BD
8118- BD BD BD BD BD BD BD BD
8120- BD BD BD BD BD BD BD BD
8128- BD BD BD 8D 00 C1 EE E6
8130- E1 EE E7 F3 E1 E4 F2 E5
8138- F3 F3 E5 A0 E9 EE A0 C8
8140- E5 F8 A0 BA A0 A0 A4 00
8148- BC A6 BE A0 EF E4 E5 F2
8150- A0 BC C3 D4 D2 CC AD D9
8158- BE A0 FA F5 ED A0 D0 F2
8160- EF E7 F2 E1 ED ED F3 F4
8168- E1 F2 F4 00
*
```

Hello

für Apple II

Ein Supermenü für jede Diskette

Allen *APPLE*-Besitzern sind die umständlichen und langen DOS-Befehle sicher schon immer ein Dorn im Auge gewesen. Hier veröffentlichen wir ein Programm, welches die wesentlichen DOS-Befehle und gleichzeitig den Catalog der zu bearbeitenden DISK zur Manipulierung anbietet.

Nach dem Start mit RUN HELLO wird der CATALOG der jeweiligen Diskette auf dem Bildschirm angezeigt und die Programme (bis zu 22) mit Buchstaben gekennzeichnet. Ein Druck auf die entsprechende Taste und das gewählte Programm kann mit RUN, EXEC oder BRUN – je nach Typ wird der passende Befehl vom Programm gewählt – gestartet werden.

Mit den Tasten 1–6 können die Funktionen LOAD, LOCK, UNLOCK, RENAME, DELETE und ein SELFSAVE des Programms auf die eingelegte Diskette aufgerufen werden. Das HELLO-Programm wird dann nach Abfrage des Programms, mit dem diese Operation durchgeführt werden soll (entspr. Buchstabe), und nach Abfrage des neuen Namens (bei Rename) den nun mit wenigen Tastendrucken zusammengestellten DOS-Befehl ausgeben.

Das Programm überprüft automatisch die Befehle auf ihre Logik, z. B. LOAD bei Texten oder der Versuch des RENAME bei einem noch geschützten Programm. Die Fehler werden akustisch und optisch gemeldet, bevor es zum DOS-ERROR kommt.

Während des Programmablaufs werden die zur Verfügung stehenden DOS-Befehle sowie LOMEN, HIMEM und FREESPACE und natürlich die Belegung der Diskette (FREE/USED) in zwei Laufschriften in Zeile 1 und 24 angezeigt.

```
?SYNTAX ERROR
ÜLIST

1 D$ =  CHR$ (4):CA$ = D$ + "CATALOG":LS$ = "TASTE DRUECKEN ODER 1=LOAD /
     2=LOCK / 3=UNLOCK / 4=RENAME / 5=DELETE / 6=SELFSAVE ......":LS$ = LS
     $ +  LEFT$ (LS$,40)
2 HO$ = "HELLO VERSION 2.2 (C) 1984 BY KLAUS SCHMIDT  >SYSTEMPARAMETER : "
     :C$(1) = "PROGRAMMSTART":C$(2) = "LOMEM":C$(3) = "HIMEM":C$(4) = "FRE
     ESPACE":D$(0) = "RUN":D$(1) = "LOAD":D$(2) = "LOCK":D$(3) = "UNLOCK":
     D$(4) = "RENAME"
3 ER$(1) = "DUMMES ZEUG!! TEXTE LADEN!!":ER$(2) = "IS' MAN SCHON UNLOCKED"
     :ER$(3) = "DOPPELTLOCKEN GEHT NICH' !!":ER$(4) = "VORHER AUFSCHLIESSE
     N !!":ER$(5) = " NUR APPLESOFT ZUR ZEIT !!":ER$(6) = "NICH' MIT MIR!
     KONZENTRIEREN SIE SICH!"
4 D$(5) = "DELETE":D$(6) = "SAVE": GOTO 9
5 C1 =  INT (CV / 8):C2 = CV - C1 * 8:P = 1024 + 128 * C2 + 40 * C1 + CH:P
     0 =  PEEK (P) - 128:P1 =  PEEK (P + 1) - 128: RETURN
6 WERT =  PEEK (AD) +  PEEK (AD + 1) * 256: RETURN
7  VTAB 24: HTAB 1: PRINT  SPC( 39);: VTAB 24: HTAB 1: RETURN
8  FOR X = 0 TO 6: CALL  - 198: NEXT : RETURN
9  TEXT : HOME : PRINT CA$:CP =  PEEK (37):AD = 103: GOSUB 6:WE(1) = WERT:
     AD = 105: GOSUB 6:WE(2) = WERT::AD = 115: GOSUB 6:WE(3) = WERT:WE(4) =
     WE(3) - WE(1)
10  FOR X = 1 TO 4:HO$ = HO$ + " " + C$(X) + "=" +  STR$ (WE(X)) + " ": NEXT
     X:BU = 0
11  FOR CV = 0 TO CP - 1:CH = 4: GOSUB 5: IF P0 = 32 THEN  GOTO 14
12  VTAB (CV + 1):SK$ = "":BU = BU + 1: IF BU = 1 THEN CA = CV
13  FOR CH = 3 TO 5: GOSUB 5:SK$ = SK$ +  CHR$ (P0): NEXT CH: POKE P - 2,6
     2: POKE P - 1,BU: POKE P,60:SK =  VAL (SK$):UD = UD + SK: HTAB 35: PRINT
     "(";SK$;")"
14  NEXT
15 HO$ = HO$ + "DISK: FREE=" +  STR$ (560 - UD) + ",USED=" +  STR$ (UD) +
     "......" +  LEFT$ (HO$,40)
16 M = M + 1: IF M =  LEN (HO$) - 40 THEN M = 1
17 N = N + 1: IF N =  LEN (LS$) - 40 THEN N = 1
18  VTAB 1: HTAB 1: PRINT  MID$ (HO$,M,40): VTAB 24: PRINT  MID$ (LS$,N,39
     );:K =  PEEK ( - 16384): IF K > 128 THEN 20
19  FOR X = 0 TO 50: NEXT : GOTO 16
20  POKE  - 16368,0:K = K - 128: IF K > 64 THEN K = K - 64:D = 0: GOTO 24
21  IF K < 49 OR K > 54 THEN ER = 6: GOTO 31
22  IF K = 54 THEN D = 6:PR$ = "HELLO": GOTO 40
23  GOSUB 7: HTAB 1:D = K - 48: VTAB 24: PRINT "(";D$(D);") - WELCHES PROG
     RAMM ";: GET K$:K =  ASC (K$) - 64
24 CH = 0:CV = CA + K - 1: GOSUB 5:LF = P0:TP = P1
25  IF TP = 84 AND D = 1 THEN ER = 1
26  IF LF = 32 AND D = 3 THEN ER = 2
27  IF K > BU THEN ER = 6
28  IF LF = 42 AND D = 2 THEN ER = 3
29  IF LF = 42 AND (D = 4 OR D = 5) THEN ER = 4
30  IF TP = 73 AND D = 0 THEN ER = 5
31  IF ER <  > 0 THEN  GOSUB 7: PRINT ER$(ER);: GOSUB 8: FOR X = 0 TO 500:
     NEXT :ER = 0: GOTO 16
32  IF TP = 84 THEN D$(0) = "EXEC"
33  IF TP = 66 THEN D$(0) = "BRUN"
34  FOR CH = 7 TO 33: GOSUB 5: IF P0 = 32 AND P1 = 32 THEN CH = 33: GOTO 3
     6
35 PR$ = PR$ +  CHR$ (P0)
36  NEXT
37  GOSUB 7: PRINT D$(D);" ";PR$;
38  IF D = 4 THEN  PRINT " , ";: INPUT NN$:PR$ = PR$ + "," + NN$: GOSUB 7
39  IF D = 5 THEN  GOSUB 8
40  GOSUB 7: PRINT D$(D);" ";PR$;
41  PRINT " O.K.?";: GET JN$: IF JN$ <  > "J" THEN PR$ = "":D$(0) = "RUN":
     GOTO 16
42  PRINT : PRINT D$;D$(D);PR$: RUN
43  END
```

Cosmische Arche

für Atari

Geschrieben auf Atari 800 XL

Ein Spiel voll Dramatik und Action

Von Ihrer Schnelligkeit hängt das Leben der Bewohner des Planeten Phaeton ab. Als Kommandant des letzten intakten Raumschiffes müssen Sie sich den Weg durch umherfliegende Asteroiden bahnen, um die Überlebenden des Planeten an Bord Ihrer Fähre zu nehmen. Ihre Energievorräte sind jedoch begrenzt, so daß Sie mit Ihnen sorgsam umgehen müssen.

Das Spiel selbst besteht aus 2 Teilen:

Asteroids
Ihre Arche steht in der Mitte des Bildschirms und Sie können in die Richtung schießen, in die Sie den Steuerknüppel bewegen. Die Asteroiden kommen immer nur aus einer von 4 Richtungen. – Je schneller Sie reagieren, desto mehr Punkte bekommen Sie.

Human Rescue
Sie versuchen mit Hilfe eines Saugstrahls die Menschen an Bord Ihrer Fähre zu nehmen. Je höher Sie dabei mit Ihrer Fähre stehen, desto mehr Punkte bekommen Sie. Vorsicht ist jedoch vor den plötzlich auftauchenden Energiewolken geboten. Ist der Rettungsversuch erfolgreich beendet, fliegen Sie an den unteren Vorsprung der Arche.

Energieverlust droht Ihrem Schiff bei folgenden Aktionen:

– ein Asteroid rammt die Arche	(5 Einheiten)
– Sie treffen den Asteroiden nicht	(5 Einheiten)
– Bewegung der Fähre	(1 Einheit)
– Sie fliegen durch eine Energiewolke	(20 Einheiten)
– der Saugstrahl trifft den Menschen nicht	(10 Einheiten)

```
0 REM ********************
1 REM *                  *
2 REM * COSMISCHE ARCHE *
3 REM *                  *
4 REM * (c)  25.12.1984 *
5 REM *                  *
6 REM * by  Jens  Berke *
7 REM *                  *
8 REM ********************
9 REM
90 SC=0:EN=0:MIS=1
100 DIM A$(44),B$(18),P(4):P(1)=14:P(3)=24:P(2)=7:P(4)=13
105 FOR A=1 TO 18:READ B:B$(A)=CHR$(B):NEXT A
106 DATA 104,104,133,207,104,133,206,104,104,168,169,0,145,206,136,208,251,96
110 CH=(PEEK(106)-8)*256:IF PEEK(203)=1 THEN 10000
120 FOR A=1 TO 44:READ B:A$(A)=CHR$(B):NEXT A
130 DATA 104,201,3,208,254,162,6,104,149,211,202
140 DATA 208,250,230,212,230,213,198,212,208,4,198
150 DATA 213,240,18,161,216,129,214,230,216,208,2
160 DATA 230,217,230,214,208,234,230,215,208,230,96
170 X=USR(ADR(A$),57344,CH,1024):RESTORE 31000
171 FOR A=33 TO 58:FOR B=1 TO 6:READ C:POKE CH+A*8+B,C:NEXT B:NEXT A
172 FOR A=16 TO 25:FOR B=1 TO 6:READ C:POKE CH+A*8+B,C:NEXT B:NEXT A:POKE CH+399
,3
180 READ A:IF A=-1 THEN POKE 203,1:GOTO 10000
```

```
190 FOR B=0 TO 7:READ C:POKE CH+A*8+B,C:NEXT B:GOTO 180
1000 GRAPHICS 28:POKE 708,148:POKE 709,248:POKE 710,188:POKE 711,54:POKE 756,CH/
256
1005 POKE 53249,100:POKE 705,248:POKE 559,46:POKE 53277,3:POKE 53256,0:POKE 5427
9,CH/256
1006 X=USR(ADR(B$),CH+640,128):POKE 623,4
1010 ME=20+SG*2:EN=EN+1000+SG*100*(SC=0)
1020 POSITION 19,8:? #6;"*"
1030 POSITION 17,9:? #6;"!";CHR$(34);"+#$"
1040 COLOR 7:PLOT 15,10:DRAWTO 22,10:POSITION 23,10:? #6;","
1050 POSITION 17,11:? #6;"%&+'("
1060 POSITION 19,12:? #6;")"
1070 FOR A=1 TO 20
1080 X=INT(RND(0)*40):Y=INT(RND(0)*19):LOCATE X,Y,Z:IF Z<>32 THEN 1080
1090 COLOR INT(RND(0)*3):PLOT X,Y:NEXT A
1091 COLOR 139:PLOT 0,20:DRAWTO 39,20
1092 POSITION 1,21:? #6;"SCORE  : ";SC:POSITION 20,21:? #6;"PRESS TRIGGER"
1093 POSITION 1,23:? #6;"ENERGY : ";EN;"              MISSION : ";MIS
1100 FOR A=1 TO 2
1110 FOR Q=1 TO 4:COLOR (A+2+1*(Q/2=INT(Q/2)))*(A=1)+32*(A=2)
1115 SOUND 0,0,0,0:GOSUB 2000:PLOT X1,Y1:DRAWTO X2,Y2:SOUND 0,96,12,5:NEXT Q:NEX
T A
1120 IF STRIG(0) THEN 1100
1129 REM *** Teil 1 (ASTEROIDS) ***
1130 POSITION 20,21:? #6;"  ASTEROIDS  ":SOUND 0,0,0,0
1150 Q=INT(RND(0)*4)+1:GOSUB 2000
1160 XA=X2:YA=Y2:XE=X1:YE=Y1:XS=XA:YS=YA
1170 XZ=SGN(XE-XA):YZ=SGN(YE-YA):P=Q
1180 REM *** Hauptschleife Teil 1 ***
1190 COLOR 32:PLOT XA,YA:XA=XA+XZ:YA=YA+YZ
1200 IF XA=XE+XZ AND YA=YE+YZ THEN EN=EN-5:GOSUB 2100:GOSUB 2200:GOTO 1150
1210 COLOR 133:PLOT XA,YA
1220 ST=STICK(0):IF ST=15 THEN 1190
1230 Q=1*(ST=11)+2*(ST=14)+3*(ST=7)+4*(ST=13):IF Q=0 THEN 1190
1235 GOSUB 2000:S=Q
1240 COLOR 3+1*(Q/2=INT(Q/2))
1250 IF P<>S THEN 1300
1260 PLOT X1,Y1:DRAWTO XA+XZ*(XA<>XE),YA+YZ*(YA<>YE):COLOR 134:PLOT XA,YA
1270 FOR A=20 TO 200 STEP 5:SOUND 0,A,0,8:NEXT A:SOUND 0,0,0,0
1275 SC=SC+50-ABS(XA-XS)*ABS(XZ)-ABS(YA-YS)*ABS(YZ)
1280 GOSUB 2200:COLOR 32:PLOT X1,Y1:DRAWTO XA,YA:ME=ME-1:IF ME=0 THEN 3000
1290 GOTO 1150
1300 PLOT X1,Y1:DRAWTO X2,Y2:SOUND 0,96,12,5:COLOR 32:PLOT X1,Y1
1310 DRAWTO X2,Y2:SOUND 0,0,0,0:EN=EN-5:GOSUB 2200:GOTO 1190
2000 IF Q/2<>INT(Q/2) THEN 2020
2010 X1=19:X2=X1:Y2=(P(Q)=7)*0+(P(Q)=13)*19:Y1=P(Q):RETURN
2020 Y1=10:Y2=Y1:X2=(P(Q)=14)*0+(P(Q)=24)*39:X1=P(Q):RETURN
2100 COLOR 134:PLOT XA-XZ,YA-YZ
2110 FOR A=0 TO 1000 STEP 20:SOUND 0,A,0,5:NEXT A:SOUND 0,0,0,0
2120 COLOR 32:PLOT XA-XZ,YA-YZ
2130 IF EN=0 THEN GOSUB 2200:GOTO 2500
2140 RETURN
2200 POSITION 10,23:? #6;EN;" ";:POSITION 10,21:? #6;SC
2210 IF EN<=0 THEN 2500
2220 RETURN
2500 POP :POSITION 3,2:? #6;"DU HAST DEINE ENERGY VERSCHWENDET.":COLOR 139:PLOT
3,3:DRAWTO 36,3
2510 SOUND 1,0,0,0:FOR A=1 TO 15000 STEP 200:SOUND 0,A,10,15-A/1000:NEXT A:SOUND
 0,0,0,0
2520 GOTO 5000
2999 REM *** Teil 2 (HUMAN RESCUE) ***
3000 HU=20+SG*3:GOSUB 2200
3010 FX=19:FY=13:MX=INT(RND(0)*38)+1
3020 POSITION 20,21:? #6;"PRESS TRIGGER"
3030 SOUND 0,50,10,5:FOR A=1 TO 200:NEXT A:SOUND 0,0,0,0
3040 IF STRIG(0) THEN 3040
3050 POSITION 20,21:? #6;"HUMAN RESCUE "
3059 REM *** Hauptschleife Teil 2 ***
3060 LOCATE FX,FY,FZ:IF FZ=10 THEN EN=EN-20:GOSUB 2200
3070 COLOR 137:PLOT FX,FY:EN=EN-1:GOSUB 2200:GOSUB 4000:GOSUB 4500
3075 IF FX=19 AND FY=13 AND HU=0 AND STICK(0)=14 THEN 6000
3080 ST=STICK(0):IF ST<>15 THEN 3090
3082 SOUND 1,0,0,0:IF STRIG(0)=0 THEN GOSUB 4600
3084 GOSUB 4000:GOSUB 4500:GOTO 3080
3090 SOUND 1,255,2,2:COLOR FZ:PLOT FX,FY
3100 FX=FX+(ST<8)*(FX<39)-(ST>8 AND ST<12)*(FX>0)
3110 FY=FY+(ST=5 OR ST=9 OR ST=13)*(FY<18)-(ST=6 OR ST=10 OR ST=14)*(FY>13)
```

```
3120 GOTO 3060
3999 REM *** Menschen bewegen ***
4000 IF MX=-1 THEN RETURN
4010 COLOR 32:PLOT MX,19:A=RND(0)
4020 MX=MX-(A<=0.5)*(MX>0)+(A>0.5)*(MX<39)
4030 COLOR 8:PLOT MX,19:SOUND 0,0,0,5:SOUND 0,0,0,0:RETURN
4499 REM *** Wolken aufbauen ***
4500 IF RND(0)<0.95-SG/100 THEN RETURN
4510 X1=INT(RND(0)*39):Y1=INT(RND(0)*6)+13:COLOR 10:PLOT X1,Y1:RETURN
4599 REM *** Saugstrahl ***
4600 POKE 53249,FX*4+48:FOR A=CH+712+(FY-13)*4 TO CH+735:POKE A,240:NEXT A
4610 SOUND 0,100,12,4
4620 IF FX=MX THEN 4640
4630 EN=EN-10:GOSUB 2200:X=USR(ADR(B$),CH+640,128):SOUND 0,0,0,0:RETURN
4640 COLOR 32:PLOT MX,19:FOR A=19 TO FY+1 STEP -1:LOCATE FX,A,C:COLOR 8:PLOT FX,
A
4650 FOR B=1 TO 20:NEXT B:COLOR C:PLOT FX,A:NEXT A
4660 SC=SC+100+(18-FY)*2:GOSUB 2200:HU=HU-1
4670 X=USR(ADR(B$),CH+640,128):SOUND 0,0,0,0:IF HU=0 THEN MX=-1:RETURN
4680 MX=INT(RND(0)*38)+1:RETURN
5000 FOR A=1 TO 40:POSITION 20,21:? #6;" PRESS START "
5010 FOR B=1 TO 20:GOSUB 5100:NEXT B
5020 POSITION 20,21:? #6;"             "
5030 FOR B=1 TO 20:GOSUB 5100:NEXT B:NEXT A:GOTO 10000
5100 IF PEEK(53279)<>6 THEN RETURN
5110 RUN
6000 POSITION 1,3:? #6;"APPLAUS...ALLE MENSCHEN SIND GERETTET."
6010 COLOR 139:PLOT 1,4:DRAWTO 38,4:SOUND 1,0,0,0
6020 FOR A=0 TO 15 STEP 0.1:SOUND 0,5,0,A:NEXT A
6030 FOR A=1 TO 300:NEXT A:FOR A=15 TO 0 STEP -0.1:SOUND 0,5,0,A:NEXT A
6040 MIS=MIS+1:SG=SG+1*(SG<3)
6050 GOTO 1000
9999 REM *** Vorspann ***
10000 GRAPHICS 0:POKE 82,0:POKE 622,255:OPEN #1,4,0,"E":POKE 710,0:POKE 752,1:?
CHR$(125)
10010 POKE 705,2:POKE 712,2:A=PEEK(560)+256*PEEK(561):POKE A+28,2:RESTORE 30000
10020 FOR A=1 TO 11
10030 READ A$,XP,L:POSITION XP,23:? A$
10040 FOR B=1 TO L:? :NEXT B:NEXT A
10050 POSITION 1,14:? "Schwierigkeitsgrad : 0":? " mit SELECT waehlen":SG=0
10060 POSITION 2,18:? "Druecke START um zu beginnen..."
10070 IF PEEK(53279)=6 THEN CLOSE #1:GOTO 1000
10080 IF PEEK(53279)<>5 THEN 10070
10090 SG=SG+1:IF SG=4 THEN SG=0
10100 POSITION 22,14:? SG
10110 FOR A=0 TO 30:SOUND 0,A,10,7:SOUND 0,0,0,0:NEXT A:GOTO 10070
30000 DATA C O S M I S C H E   A R C H E,5,10
30010 DATA (c)  JB'84  25.12.1984,9,10
30020 DATA GRAPHIC UND PROGRAMM,10,1
30030 DATA VON,19,1
30040 DATA JENS BERKE,15,1
30045 DATA FINKENHOF  7,14,1
30050 DATA 3070  NIENBURG,13,10
30060 DATA TEIL 1 : ASTEROIDS,1,1
30070 DATA ABWEHR DER UMHERFLIEGENDEN ASTEROIDEN.,1,3
30080 DATA TEIL 2 : HUMAN RESCUE,1,1
30090 DATA RETTUNG DER LETZTEN UEBERLEBENDEN.,1,13
31000 DATA 48,204,136,168,204,204,240,204,160,160,204,240,60,192,128,128,192,60
31010 DATA 240,204,136,136,204,240,252,192,128,160,192,252,252,192,128,160,192,1
92
31020 DATA 60,192,128,136,204,60,204,204,136,168,204,204,252,48,32,32,48,252
31030 DATA 12,12,8,136,204,48,204,204,160,160,204,204,192,192,128,128,192,252
31040 DATA 204,252,136,136,204,204,204,252,168,168,204,204,48,204,136,136,204,48
31050 DATA 240,204,136,160,192,192,48,204,136,136,204,60,240,204,136,160,240,204
31060 DATA 60,192,128,32,12,240,252,48,32,32,48,48,204,204,136,136,204,252
31070 DATA 204,204,136,136,204,48,204,204,136,136,252,204,204,204,32,32,204,204
31080 DATA 204,204,136,32,48,48,252,12,32,128,192,252,168,136,136,136,136,168
31090 DATA 32,160,32,32,32,168,168,8,8,168,128,168,168,8,32,8,8,160,136,136,136,
170,8,8
31100 DATA 168,128,128,168,8,168,168,128,168,136,136,168,168,8,8,40,32,32
31110 DATA 168,136,136,168,136,168,168,136,168,8,8,168
32000 DATA 1,0,0,0,0,1,5,22,90
32010 DATA 2,1,5,22,90,106,170,170,170
32020 DATA 3,64,80,148,165,169,170,170,170
32030 DATA 4,0,0,0,0,64,80,148,165
32040 DATA 5,90,22,5,1,0,0,0,0
32050 DATA 6,170,170,170,106,90,22,5,1
```

```
32060 DATA 7,170,170,170,169,165,148,80,64
32070 DATA 8,165,148,80,64,0,0,0,0
32080 DATA 9,105,105,105,105,105,40,0,0
32090 DATA 10,0,0,40,105,105,105,105,105
32100 DATA 11,170,170,170,170,170,170,170,170
32105 DATA 12,85,85,40,40,40,40,85,85
32110 DATA 64,0,0,48,0,0,0,0,0
32120 DATA 65,0,0,0,0,12,0,0,0
32130 DATA 66,0,0,0,0,0,0,3,0
32140 DATA 67,0,0,170,170,170,170,0,0
32150 DATA 68,40,40,40,40,40,40,40,40
32160 DATA 69,0,48,236,255,63,59,12,0
32170 DATA 70,3,204,0,44,11,48,204,3
32180 DATA 71,85,85,42,42,42,42,85,85
32190 DATA 72,32,32,0,252,48,48,204,204
32200 DATA 73,0,20,85,170,60,170,85,20
32210 DATA 74,0,48,252,255,255,255,60,0
32220 DATA 75,255,255,0,0,0,0,0,0
32767 DATA -1
```



ATLANTIS:

für Atari

Nach einem Erdbeben ist der 7. Kontinent, ATLANTIS, wieder aus dem Meer emporgekommen.

Geschrieben auf dem Atari 800 XL.

Doch anstatt Ruinen sieht man mit Kuppeln überdachte Städte, Kraftwerke, Brücken u.s.w. Die Bewohner von ATLANTIS haben also überlebt. Ihre alten Feinde jedoch, die Klagonier, haben auch nicht geschlafen und schicken sofort ihre Flotte aus, den Kontinent in eine Wüste zu verwandeln. Sie als Kommandant müssen nun mit 3 superschweren Geschützen die Feinde aufhalten. Ob Ihnen das gelingt? Man wird sehen...

Mit der linken und rechten Kanone schießt man, indem der Steuerknüppel nach links bzw. rechts bewegt und dann der Feuerknopf gedrückt wird. Für die mittlere Kanone müssen Sie den Steuerknüppel in der neutralen Stellung lassen.

Punktewertung:

Für die langsamen Raumschiffe gibt es 20, für die schnellen 50 Punkte.

Die linke und rechte Kanone können aufgrund einer speziellen Panzerung nicht zerstört werden. Hat man die mittlere Kanone verloren, so muß eine bestimmte Punktzahl hinzugewonnen werden, um eine neue zu erhalten. Die Raumschiffe können nur getroffen werden, wenn der Schuß sie von unten erreicht. Leider wurde die linke Kanone fehlerhaft gebaut und kann aus diesem Grunde die schnellen Feinde nicht treffen.

Das Spiel ist zu Ende, wenn alle zerstörbaren Bauwerke vernichtet worden sind. Es kann übrigens durch gleichzeitiges Drücken von CONTROL und 1 unterbrochen und durch eine nochmalige Betätigung fortgesetzt werden.

Hat man vorher schon ein anderes Spiel gespielt, so sollte vor dem Programmstart von ATLANTIS der Befehl POKE 203,0 eingegeben werden.

Bei der Eingabe des Programms sollte besonders auf die Richtigkeit der PRINT-Befehle geachtet werden.

Listing:

```
0 REM ********************
1 REM *                  *
2 REM * (c) 19.8.1984 by *
3 REM *                  *
4 REM *    Jens Berke    *
5 REM *                  *
6 REM ********************
7 REM
```

```
100 DIM A$(44),RX(4),RY(4),CO(4),SL(39),WG(39):CH=(PEEK(106)-8)*256:IF PEEK(203)
<>0 THEN 11000
110 FOR A=1 TO 44:READ B:A$(A)=CHR$(B):NEXT A
120 DATA 104,201,3,208,254,162,6,104,149,211,202,208,250,230,212,230,213,198,212
,208,4,198
130 DATA 213,240,18,161,216,129,214,230,216,208,2,230,217,230,214,208,234,230,21
5,208,230,96
140 Q=USR(ADR(A$),57344,CH,895):RESTORE 32000
150 READ B:IF B=-1 THEN POKE 203,1:GOTO 11000
160 FOR A=0 TO 7:READ C:POKE CH+B*8+A,C:NEXT A:GOTO 150
200 RESTORE 210:FOR A=0 TO 39:READ B:SL(A)=B:NEXT A
210 DATA 13,13,14,15,15,16,16,16,15,16,17,17,17,17,17,17,16,15,14,15,16,16,16,16
,16,17,17,17,17,17,17,17
220 DATA 18,18,18,17,15,14,13,13
230 FOR A=0 TO 39:READ B:WG(A)=B:NEXT A:RETURN
240 DATA 0,0,0,2,2,0,1,0,1,0,0,3,3,3,3,0,1,0,4,0,0,3,3,3,3,0,5,5,5,5,5,0,2,2,0
,1,0,0,0
1000 GRAPHICS 28:POKE 752,1:POKE 82,0:POKE 708,54:POKE 710,164:POKE 709,252:POKE
 711,150:POKE 756,CH/256
1010 DL=PEEK(560)+256*PEEK(561)
1020 POKE DL+28,2:POKE DL+6,2
1030 POSITION 0,14:? #6;CHR$(219);CHR$(220):POSITION 38,14:? #6;CHR$(221);CHR$(2
22);
1040 ? #6;"%%#             &                    $%%";
1050 ? #6;"%%%*+    ";CHR$(223);"          $%#                 ";CHR$(223);"%%%";
1060 ? #6;"%%%%%#'$%#          '%%%#";CHR$(172);CHR$(173);CHR$(174);CHR$(175);"
    %%%%";
1070 ? #6;"%%%%%%%%%#,-./$%%%%%%%%%%#()()()   $%%%%";
1080 ? #6;"%%%%%%%%%%%%%%%%%%%%%%%%%%#  $%##*+%%%%%";
1090 ? #6;"%%%%%%%%%%%%%%%%%%%%%%%%%%%#$%%%%%%%%%%%";
1100 COLOR 37:PLOT 0,21:DRAWTO 39,21:PLOT 0,22:DRAWTO 39,22
1110 POSITION 4,23:? #6;"A T L A N T I S    BY JENS BERKE"
1120 POSITION 12,1:? #6;"SCORE : 0"
1130 SC=0:AR=0:AG=10:KF=1
1140 GOSUB 200:SOUND 3,255,12,1
2000 GOSUB 5000:GOSUB 3000:IF AR=0 THEN 2000
2010 N=N+1:IF N>AR THEN N=1
2020 COLOR 32:PLOT RX(N),RY(N):RX(N)=RX(N)+1+(CO(N)=34)
2030 IF RX(N)=40 THEN RX(N)=0:RY(N)=RY(N)+2:IF RY(N)>11 THEN RY(N)=5
2040 COLOR CO(N):PLOT RX(N),RY(N)
2050 SZ=SZ+RND(0)/3+SG/5:IF SZ<10 THEN 2000
2060 SZ=0:A=INT(RND(0)*AR)+1
2070 COLOR 60:PLOT RX(A),RY(A)+1:DRAWTO RX(A),SL(RX(A)):SOUND 0,155,12,4
2080 GOSUB 7000:SOUND 0,0,0,0
2090 COLOR 32:PLOT RX(A),RY(A)+1:DRAWTO RX(A),POS:GOTO 2000
3000 IF SF=1 THEN GOSUB 4000:RETURN
3010 IF STRIG(0) THEN RETURN
3020 SO=6:SF=1:ST=STICK(0):IF ST>12 AND KF THEN 3080
3030 IF ST<8 THEN 3070
3040 IF ST>8 AND ST<12 THEN 3060
3050 SF=0:RETURN
3060 ZX=-1:SX=1:SY=13:CO=6:GOTO 3000
3070 ZX=1:SX=38:SY=13:CO=7:GOTO 3000
3080 ZX=0:SX=18:SY=14:CO=62:GOTO 3000
4000 COLOR 32:PLOT SX,SY:SX=SX-ZX:SY=SY-1:SO=SO+2:SOUND 2,SO,0,2
4010 IF SY=4 THEN SF=0:SOUND 2,0,0,0:RETURN
4020 LOCATE SX,SY,Z:IF Z>32 AND Z<35 THEN SOUND 2,0,0,0:GOTO 6000
4030 COLOR CO:PLOT SX,SY:RETURN
5000 IF AR=4 THEN RETURN
5010 RZ=RZ+RND(0)+SG/5:IF RZ<7 THEN RETURN
5020 RZ=0:AR=AR+1:RX(AR)=0:RY(AR)=5:CO(AR)=33:IF RND(0)<(SG+1)/10 THEN CO(AR)=34
5030 RETURN
6000 FOR A=1 TO AR:IF RX(A)=SX AND RY(A)=SY THEN 6020
6010 NEXT A
6020 COLOR 64:PLOT RX(A),RY(A):FOR B=0 TO 100 STEP 5:SOUND 0,B,0,10:NEXT B:SOUND
 0,0,0,0
6030 COLOR 32:PLOT RX(A),RY(A)
6040 FOR B=A TO AR-1:RX(B)=RX(B+1*(B<4)):RY(B)=RY(B+1*(B<4)):CO(B)=CO(B+1*(B<4))
:NEXT B
6050 AR=AR-1:SC=SC+20+30*(Z=34):SG=SG+0.02*(SG<>5):GOSUB 9000:SF=0:RZ=0
6060 IF KF=0 THEN SCF=SCF+20+30*(Z=34):IF SCF>(SG*2+1)*100 THEN GOSUB 10000
6070 RETURN
7000 POS=SL(RX(A)):ON WG(RX(A)) GOTO 7100,7200,7300,7400,7500
7010 RETURN
7100 C=RX(A):COLOR 64:PLOT C,POS+1:SOUND 0,150,10,2
7110 FOR B=0 TO 100 STEP 2:SOUND 1,B,0,8:NEXT B:SOUND 0,0,0,0:SOUND 1,0,0,0
```

```
7120 COLOR 32:PLOT C,POS+1:D=0:GOTO 8000
7200 IF RX(A)<5 THEN C=3:GOTO 7220
7210 C=33
7220 COLOR 64:PLOT C,POS+1:PLOT C+1,POS+1
7230 FOR B=1 TO 20:SOUND 0,RND(0)*100,0,6:NEXT B:SOUND 0,0,0,0
7240 COLOR 32:PLOT C,POS+1:PLOT C+1,POS+1:D=1:GOTO 8000
7300 IF RX(A)<15 THEN C=11:GOTO 7320
7310 C=21
7320 COLOR 189:PLOT C,POS+1:DRAWTO C+3,POS+1
7330 FOR B=0 TO 50 STEP 2:SOUND 0,B,0,6:SOUND 1,B,12,6:NEXT B:SOUND 0,0,0,0:SOUN
D 1,0,0,0
7340 COLOR 32:PLOT C,POS+1:DRAWTO C+3,POS+1:D=3:GOTO 8000
7400 C=18
7410 COLOR 189:PLOT C,POS+1:FOR B=255 TO 0 STEP -15:SOUND 0,B,0,8:NEXT B:SOUND 0
,0,0,0
7420 COLOR 32:PLOT C,POS+1:D=0:KF=0:SCF=0:GOTO 8000
7500 C=26
7510 COLOR 64:PLOT C,POS+1:DRAWTO C+5,POS+1
7520 FOR B=0 TO 100 STEP 5:SOUND 0,B+90,10,2:SOUND 1,B,12,6:NEXT B:SOUND 0,0,0,0
:SOUND 1,0,0,0
7530 COLOR 32:PLOT C,POS+1:DRAWTO C+5,POS+1:D=5:GOTO 8000
8000 FOR B=C TO C+D:SL(B)=SL(B)+1:WG(B)=0:NEXT B:AG=AG-1
8010 IF AG=0 THEN POP :GOTO 12000
9000 POSITION 20,1:? #6;SC:RETURN
10000 COLOR 38:PLOT 18,15:WG(18)=4:SL(18)=14:KF=1:AG=AG+1
10010 FOR B=50 TO 0 STEP -3:SOUND 0,B,10,5:SOUND 1,B+1,10,5:NEXT B:SOUND 0,0,0,0
:SOUND 1,0,0,0
10020 RETURN
11000 GRAPHICS 4:POKE 712,252:POKE 710,252:POKE 709,0:POKE 708,38:POKE 752,1:POK
E 82,0:? CHR$(125)
11010 SC=PEEK(88)+256*PEEK(89)
11020 COLOR 1:PLOT 0,15:DRAWTO 79,15:DRAWTO 79,24:DRAWTO 0,24:DRAWTO 0,15
11030 RESTORE 11130:FOR A=0 TO 7:READ C:FOR B=0 TO 7
11040 POKE SC+161+A+B*10,PEEK(57344+C*8+B):SOUND 0,7-B,10,8:NEXT B:SOUND 0,0,0,0
:NEXT A
11050 ? "Copyright (c)  August 1984 by Jens Berke":? "      Select skill level :
NOVICE"
11060 ? "       then press START";
11070 A$="NOVICE  SKILLED ADVANCEDEXPERT  ":SG=0:FOR A=1 TO 4:? CHR$(31);:NEXT A
:? CHR$(28);
11080 IF PEEK(53279)=6 THEN 1000
11090 IF PEEK(53279)<>5 THEN 11080
11100 SG=SG+1:IF SG=4 THEN SG=0
11110 POSITION 26,22:? A$(SG*8+1,SG*8+8);"   ";
11120 FOR A=1 TO 11:? CHR$(30);:NEXT A:FOR A=1 TO 50:NEXT A:GOTO 11080
11130 DATA 33,52,44,33,46,52,41,51
12000 SOUND 0,0,0,0:SOUND 1,0,0,0:SOUND 2,0,0,0:SOUND 3,0,0,0
12010 POKE DL+14,2:POKE DL+16,2:POSITION 11,9:? #6;"G A M E    O V E R"
12020 FOR A=0 TO 160:SOUND 0,A,A,6:NEXT A:SOUND 0,0,0,0
12030 ? #6:? #6;"   DRUECKE START FUER EIN NEUES SPIEL"
12040 IF PEEK(53279)=6 THEN RUN
12050 GOTO 12040
32000 DATA 1,0,0,0,80,69,20,0,0
32010 DATA 2,0,0,20,65,85,20,0,0
32020 DATA 3,0,240,240,240,252,252,255,255
32030 DATA 4,3,15,15,15,63,255,255,255
32040 DATA 5,255,255,255,255,255,255,255,255
32050 DATA 6,0,40,40,40,130,170,130,170
32060 DATA 7,0,32,152,32,152,32,152,32
32070 DATA 8,0,1,1,4,4,16,64,85
32080 DATA 9,64,80,80,68,68,65,64,85
32090 DATA 10,0,0,5,6,5,22,21,21
32100 DATA 11,0,0,80,144,80,148,84,84
32110 DATA 12,0,0,2,8,32,33,129,143
32120 DATA 13,10,160,1,1,49,49,125,255
32130 DATA 14,160,10,0,12,12,172,15,255
32140 DATA 15,0,0,128,32,8,72,66,242
32150 DATA 28,40,40,40,40,40,40,40,40
32160 DATA 30,40,60,40,60,40,60,40,60
32170 DATA 32,0,4,65,16,4,64,17,0
32180 DATA 59,0,0,0,0,3,3,15,51
32190 DATA 60,48,48,192,192,15,51,255,51
32200 DATA 61,12,12,3,3,48,204,255,204
32210 DATA 62,0,0,0,0,192,192,240,204
32220 DATA 63,0,12,29,12,29,12,29,12
32230 DATA 29,0,32,2,192,12,2,128,32,-1
```

Punktefresser

für VC-20 ohne Erweiterung



Geschicklichkeitsspiel für eine Person

Nachdem sich das Spielfeld aufgebaut hat, ist es Ihre Aufgabe in einem Labyrinth alle verfügbaren Punkte zu „fressen". Ist Ihnen das gelungen, erhalten Sie ein Zusatzspiel und der Schwierigkeitsgrad des Spieles steigt.
Innerhalb des Labyrinths tauchen jedoch Monster auf, deren Aufgabe es ist Sie zu töten. Sollten Sie also von einem Monster gefressen werden, erscheint auf Ihrem Bildschirm ein Kreuz und eine Melodie erklingt. Danach baut sich das Spielfeld wieder auf und das Spiel beginnt von vorn. Gesteuert wird mit Joystick. Los geht's und „Guten Appetit"!

```
70 REM===============
72 REM=  PUNKTE -    =
74 REM=              =
76 REM=    FRESSER   =
78 REM===============
80 PRINT"J":PRINT"■■■      PUNKTEFRESSER     ■■"
85 POKE36879,90
90 PRINT"■■SIE MUESSEN VERSUCHEN ALLE PUNKTE ZU FRESSENUM DAFUER PUNKTE ZU    ERH
ALTEN."
95 PRINT"■ABER VORSICHT DIE MON-STER WOLLEN SIE TOETEN"
99 FORU=1TO8000:NEXT
100 FORI=0TO34:READX:POKE828+I,X:NEXT
110 DATA169,127,141,34,145,173,32,145,41,128,133,251,169,255,141,34
120 DATA 145,173,31,145,41,28,24,101,251,133,251,173,31,145,41,32
130 DATA 133,252,96
140 PRINT"F■■■■■■■■     PUNKTEFRESSER      ■■"
150 PRINT   "■■■■STEUERUNG MIT JOYSTICK■■■■VIEL GLUECK!!"
160 FORX=0TO7:POKE7168+32*8+X,0:NEXT
170 FORX=0TO7:READA:POKE7168+46*8+X,A:NEXT
180 FORX=0TO7:READA:POKE7168+X,A:NEXT
190 FORX=0TO7:READA:POKE7168+42*8+X,A:NEXT
200 FORX=0TO7:READA:POKE7168+1*8+X,A:NEXT
210 FORX=0TO7:READA:POKE7168+2*8+X,A:NEXT
220 DATA0,0,0,24,24,0,0,0
230 DATA0,90,0,90,90,0,90,0
240 DATA219,219,90,126,24,60,126,195
250 DATA24,24,126,189,189,36,36,36
260 DATA24,24,126,126,24,24,24,24
270 FORZZ=1TO3000:NEXT
280 POKE36869,255
290 PRINT"J"
300 L=22:Z=7680:F=200:O=46:I=46:POKE36878,15:K=1:J=0:W=0
310 POKE36879,10
320 PRINT"■"
330 PRINT"■■@@@@@@@@@@@@@@@@@@@@@@"
340 FORT=0TO5
350 PRINT"@....................."
360 PRINT"@....................."
370 PRINT"@....................."
380 NEXTT
390 PRINT"@....................."
400 PRINT"@@@@@@@@@@@@@@@@@@@@@@"
```

```
410 FORZZ=0TO20:POKE7701+ZZ*L,0 :NEXT
420 FORZZ=1TO100
430 ZY=INT(RND(1)*439)+7702:IFPEEK(ZY-23)=0 ORPEEK(ZY+23)=0 ORPEEK(ZY-21)=0 THEN
430
440 IFPEEK(ZY+21)= 0ORPEEK(ZY)=0THEN430
450 POKEZY,0
460 NEXT ZZ
470 PRINT"■"
480 FORZZ=0TO20:POKE8142+ZZ,32:NEXT
490 POKE36875,200:FORFG=1TO200:NEXT:POKE36875,0
500 IFJ>0THEN580
510 X=19
520 Y=19
530 P=Z+X+L*Y
540 D=INT(10*RND(1)):E=INT(10*RND(1)):G=Z+D+L*E
550 IFPEEK(G)=0ORPEEK(G)=32ORPEEK(G)=1THEN540
560 M=INT(10*RND(1)):N=INT(10*RND(1)):T=Z+M+L*N
570 IFPEEK(T)=0ORPEEK(T)=32ORPEEK(T)=1THEN560
580 POKEP,1:POKEG,42
590 A=X:B=Y:Q=P
600 W=W+1:IFW=VTHENW=0:FORS=1TO100:NEXT:GOTO620
610 GOSUB730:GOSUB990
620 V=INT((K/2)/F)+2
625 SYS828
630 IFPEEK(251)=140THENX=X-1:GOTO670
640 IFPEEK(251)=28THENX=X+1:GOTO670
650 IFPEEK(251)=148THENY=Y+1:GOTO670
660 IFPEEK(251)=152 THENY=Y-1
670 P=Z+X+L*Y
680 IFPEEK(P)=0THENX=A:Y=B:P=Q:GOTO590
690 IFPEEK(P)=42THENPOKEQ,32:GOTO1090
700 IFPEEK(P)=46THENPOKE36875,230:J=J+1:K=K+.5:PRINT"▒]■";J*10;"▌":POKE36875,0
710 IFJ=278ORJ=100556THENJ=J+100000:GOTO320
720 POKEQ,32:POKEP,1:GOTO590
730 H=G:DK=D:EK=E
740 IFD=XTHEN830
750 IFD<XTHEND=D+1:GOTO770
760 IFD>XTHEND=D-1
770 G=Z+D+L*E
780 IFPEEK(G)=1THENPOKEH,I:GOTO1090
790 IFPEEK(G)=0ORPEEK(G)=42THENG=H:D=DK:E=EK:GOTO830
800 POKEH,I
810 I=PEEK(G)
820 POKEG,42:RETURN
830 IFE<YTHENE=E+1:GOTO850
840 IFE>YTHENE=E-1
850 G=Z+D+L*E
860 IFPEEK(G)=1THENPOKEH,I:GOTO1090
870 IFPEEK(G)=0ORPEEK(G)=42THENE=EK:G=H:RETURN
880 POKEH,I
890 I=PEEK(G)
900 POKEG,42:RETURN
910 IFM<XTHENM=M+1:GOTO930
920 IFM>XTHENM=M-1
930 T=Z+M+L*N
940 IFPEEK(T)=1THENPOKER,O:GOTO1090
950 IFPEEK(T)=0ORPEEK(T)=42THENM=MK:T=R:RETURN
960 POKER,O
970 O=PEEK(T)
980 POKET,42:RETURN
990 R=T:MK=M:NK=N
1000 IFN=YTHEN910
1010 IFN<YTHENN=N+1:GOTO1030
1020 IFN>YTHENN=N-1
1030 T=Z+M+L*N
1040 IFPEEK(T)=1THENPOKER,O:GOTO1090
1050 IFPEEK(T)=0ORPEEK(T)=42THENN=NK:T=R:GOTO910
1060 POKER,O
1070 O=PEEK(T)
1080 POKET,42:RETURN
1090 POKEP,2
1100 PRINT"]▒ERGEBNIS:";J*10
1110 FORT=255TO200STEP-.1
1120 POKE36876,T
1130 NEXTT
1140 POKE36876,0
1150 GOTO300
```

Star-Wars

Für VC-20 ohne Erweiterung

Dieses für BASIC-Verhältnisse schnelle Reaktionsspiel basiert in Graphik und Idee zum größten Teil auf dem Science-fiction-Film „STAR-WARS".

Sie steuern ein Raumschiff, am oberen Bildschirmrand dargestellt, und haben die Aufgabe, mit ihren 3 Raumschiffen bzw. 3 Leben solange wie möglich heil durch das mit todbringenden Minen bestückte Asteroidenfeld zu fliegen. Allerdings wird diese Aufgabe dadurch erschwert, daß sie während den schwierigen Ausweichmanövern auch noch die feindlichen UFO-Geschwader zur Strecke bringen sollen. Für jedes getroffene UFO gibt's 50 Punkte.

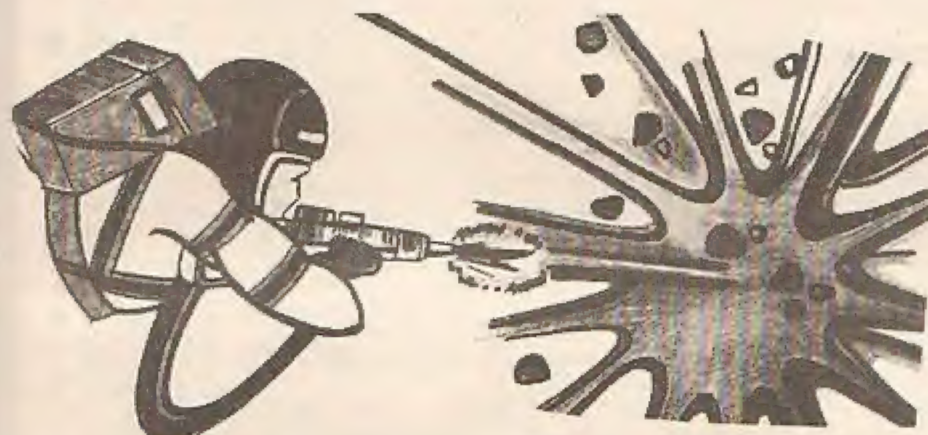
Sollten Sie zufälligerweise auf einen Meteoriten schießen, wird dieser nicht nur ganzgelassen, sondern auch ihre Punktzahl schrumpft um volle 10 Punkte ab. Auch die gefährlichen Alpha-Minen sollten Sie nicht als Objekt für Ihre unfreiwilligen Schießübungen benutzen: Es könnte Ihr letzter Schuß sein!!!

Gesteuert wird Ihr Raumschiff mit den Tasten:

O = links   rechts = P   F = Feuer

Bei Kollision mit einem Meteoriten bzw. einem UFO oder einer Mine verlieren Sie ein Schiff.

Wie schon im Listing bemerkt, kann das Meteoritenfeld selber gestaltet werden. Hierbei ist L = Meteorit, S = Mine und B = UFO. Allerdings ist die Anzahl der A-Strings auf 20 dimensioniert.



```
0 POKE52,28:PRINT"□":POKE36879,8:P=36879
1 PRINT"[illegible]DATAS WERDEN GELESEN!"
10 POKE56,28
20 DATA8,8,8,73,85,119,127,20,0,20,20,42,20,73,36,129
22 DATA129,153,189,231,189,153,129,129,37,73,34,148,61,40,82,40
23 DATA0,0,0,0,4,0,0,0,126,129,157,161,161,157,129,126
25 DATA127,128,143,144,136,132,252,128,255,0,255,68,74,95,81,0
27 DATA254,1,249,53,41,37,35,1,128,193,170,170,170,148,128,127
29 DATA0,255,146,147,146,98,0,255,1,143,81,201,69,61,1,254
31 DATA8,60,126,126,54,8,36,73,127,145,17,57,81,17,33,0
33 DATA255,69,229,37,21,21,23,0,253,34,64,88,68,36,24,0
34 DATA0,0,65,65,65,0,0,0,20,127,119,85,73,8,8,8
35 DATA129,36,73,20,42,20,20,0,129,66,24,244,47,24,66,129,999
40 RESTORE
50 I=0
60 READA:IFA=999THEN80
70 POKE7168+I,A:I=I+1:GOTO60
80 FORF=7424TO7432:POKEF,0:NEXTF:CLR
81 GOSUB90
82 GOSUB91
83 GOSUB93
```

```
84 GOTO100
90 FORG=7680TO8185:POKEG,32:NEXTG:RETURN
91 POKE36869,255:RETURN
93 FORS=1TO30:X=INT(RND(1)*500)+7680:POKEX,4:NEXTS:RETURN
100 PRINT"■■■■■■■■■■■■■■■■■■■FGH■■■■IJK"
110 PRINT"■■■■■■■■■BY■■MNO■-SOFT",,"■■■■LTD.■"
111 FORU=1TO1000:NEXTU:P=36879:POKEP-1,10
112 V=0:GOSUB126:FORT=1TO200:NEXTT
113 FORL=1TO22:V=V-22:POKEP-4,240:FORM=1TO30:NEXTM:POKEP-4,0:POKEP-3,240:FORM=1T
O10:NEXT
114 POKEP-3,0:POKE8160+V,0:POKE8182+V,1:FORM=1TO10:NEXTM:POKE8182+V,32:NEXTL
115 POKEP-3,0:POKEP-4,0
116 POKEP-2,220:FORL=15TO0STEP-.5:POKEP-1,L:FORM=1TO150:NEXTM:NEXTL:POKEP-1,0:PO
KEP-2,0
117 GOTO135
126 V=0:FORJ=200TO222:POKEP-2,J:POKE7704+V,2:POKE7682+V,32:V=V+22
127 FORT=1TO20:NEXTT:NEXTJ:V=0:POKEP-2,0:RETURN
135 GOSUB90:POKE36869,242
136 PRINT"■"
137 PRINT"■■■■■■■|■■TAR-OARS■|"
138 PRINT"■■■■■|──────────┘"
140 PRINT"■■■IE SIND DER ¬ILOT    EINES ¬ATROULLIEN-    ■CHIFFES DER _EBELLEN-_L
OTTE.■"
141 PRINT"■■■IE HABEN DIE ■UFGABE ERHALTEN, ¬ATROULLIE  DURCH EINEN VOM ¬MP-  ERI
UM BESETZTEN"
142 PRINT"■■_AUM-■EKTOR ZU FLIEGEN       ■  !!!! "
143 GOSUB200:POKEP-1,15:GOSUB499
144 GOSUB290:POKEP,10
145 PRINT"■■■■■IE STEUERN DEN         ■-OING-_LUEGLER        MIT DEN |ASTEN:"
146 PRINT"■■■ ■[┐]=LINKS _ECHTS=[┐]■       ■■ [_]=_EUER":P=36879:POKEP-1,15
147 FORL=130TO254:POKEP-3,L:POKEP-2,L:POKEP-4,L:NEXTL:POKEP-3,0:POKEP-2,0:POKEP-
4,0
148 GOSUB200:FORJ=38TO82STEP.5:POKE36865,J:POKEP-3,J+170:NEXTJ:POKE36865,38
149 POKE36869,240:POKEP,8:POKEP-3,0
150 GOTO400
200 PRINT"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■TASTE!"
201 GETD$:IFD$=""THEN201
202 RETURN
290 GOSUB90:POKEP,8
291 POKE36869,255
292 PRINT"■■"
293 GOSUB93
294 PRINT"■■■■■■■■■■■■FGH■■■■IJK"
295 PRINT"■■■■■■ ZERSTOEREN SIE DIE    FEINDLICHEN            TIE-JAEGER  ■
[■■■]"
296 PRINT"■■■ ABER TREFFEN SIE      BLOSS NICHT DIE HOCH- EXPLOSIVEN @-MINEN■[■■
S■■]"
297 PRINT"■■■ ES KOENNTE IHR         LETZTER SCHUSS SEIN           !!!!!!"
299 GOSUB200:POKE36869,242:PRINT"■":RETURN
400 PRINT"■"
401 PRINT"■■ BITTE WARTEN SIE NUN!  DER VC-20 LAEDT       ■   ■ ■STAR-WARS 2■"
402 FORH=1TO1700:NEXTH
403 CLR:POKE198,3:POKE632,147:POKE633,131
404 END
499 GOSUB90:PRINT"■":POKE36869,255:GOSUB93:POKE8130,0:POKE8152,1:POKE7756,2:FORK
=1TO30
500 POKE38872,INT(RND(1)*9):POKEP-2,INT(RND(1)*100)+100:NEXTK:POKEP-2,0
501 C=8108:FORH=1TO15:D=D-22:POKEC+D,16:POKEP-3,250:FORG=1TO20:NEXTG:POKEC+D,32:
POKEP-3,0:NEXTH
502 POKEP-2,210:POKE7756,3:FORT=15TO0STEP-.1:POKEP-1,T:NEXTT
503 RETURN
```

Programm Teil 2

```
0 REM** (C) BY FROGSOFT LTD.**
1 REM** LUTZ BLOMBERG 1984  **
2 PRINT"■":P=36879:POKE808,127:REM"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
3 SH=3:F=30720:M=7702:I1=0:I2=17:X1=1:X2=18
4 J=2:Q=12:O=16:L=11:GOTO20000
5 GOSUB130:S=S+1:GOSUB115:GOSUB135:GOSUB140:GOSUB145:GOSUB120:FORT=1TO5:POKEP-T,
0:NEXTT
6 GOSUB122:GOSUB130:GOTO499
```

```
115 POKE36869,255:RETURN
120 FORT=1TO3:FORR=1TO3:FORY=128TO250STEP3:POKEP-1,5+(R*2):POKEP-6+R,Y:NEXTY,R,T
:RETURN
122 POKEP-1,15:POKEP-2,177:FORTT=15TO0STEP-1:FORY=1TO99:NEXTY:POKEP-1,0
123 FORR=1TO15:NEXTR:POKEP-1,TT:NEXTTT:RETURN
130 FORY=7680TO8185:POKEY,32:NEXTY:RETURN
135 FORT=1TO30:K=(INT(RND(1)*500)+7680):POKEK,4:NEXTT:RETURN
140 PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒FGH▒▒▒▒IJK"
141 PRINT"▒▒▒▒▒▒E▒▒BY▒▒MNO▒-SOFT",,"▒▒▒▒LTD.▒":RETURN
145 POKEP-1,15:FORT=1TO50:PRINT"▒▒▒▒▒▒▒▒▒▒SHIPS";SH:POKEP-4,250
146 PRINT"▒▒▒▒▒▒▒▒▒▒SHIPS";SH:POKEP-4,0:NEXTT:RETURN
499 IFSH<>3THEN518
500 TI$="000000":DIMA$(20)
502 A$(1)=" L                    "
503 A$(2)=" ▒▒▒             L "
504 A$(3)="     L      L            "
505 A$(4)="                         "
506 A$(5)="          ▒▒▒    L"
507 A$(6)="               L    L "
508 A$(7)="               L   ▒B"
509 A$(8)=" ▒▒▒    L    L    "
510 A$(9)="     L          "
511 A$(10)="          ▒▒▒      L "
512 A$(11)="  L   ▒▒▒     L     "
513 A$(12)="  L                 L"
514 A$(13)="     L    ▒▒▒     "
515 A$(14)="     ▒▒▒ L    L       "
516 POKEM+L,32
517 B=INT(RND(1)*14)+1:PRINT"▒▒▒▒▒▒▒▒",A$(B);
518 IFPEEK(7702+L)<>32THENGOTO650
519 POKEM+L,12:POKEM+L+F,6:POKEM+L-22,X2:POKEM+L-22+F,INT(RND(1)*3)+1:POKEP-1,10
520 POKEP-2,INT(RND(1)*6)+180
521 PE=PEEK(197):IFPEEK(7702+L+22)<>32THENGOTO650
522 IFPE=52THENL=L-1:IFL<1THENL=1
523 IFPE=13THENL=L+1:IFL>21THENL=21
524 IFPE=42THEN559
526 GOTO516
559 C=0
560 KL=255:FORT=M+L+44TOM+L+18*22STEP22:POKET,0:POKET-22,32:POKET+F,INT(RND(1)*3
)+6
561 POKEP-3,KL:KL=KL-2:IFPEEK(T+22)<>32THENPOKEP-3,0:GOTO680
562 NEXT:POKEP-3,0:POKET-22,32:GOTO516
563 REMEMBER TO  BUY       THE NEW FROGSOFT          GAMES!
650 POKEM+L,3:POKEM+L+F,7:GOSUB122:POKEM+L,32:GOTO1000
680 TRE=PEEK(T+22):IFTRE=2THENSC=SC+50
682 IFTRE=19THENGOTO10000
684 IFTRE=12THENSC=SC-10:POKET,32:GOTO516
690 POKET+22,3:POKET+22+F,7:POKET,32:GOSUB800:POKET+22,32
700 GOTO516
800 POKEP-2,222:FORGV=15TO0STEP-.2:POKEP-1,GV:NEXT:POKEP-2,0:RETURN
999 REMEMBER TO BUY THE NEW FROGSOFT GAMES!
1000 SH=SH-1:IFSH=0THEN1002
1001 POKEP-2,0:POKEP-3,0:GOTO5
1002 GOSUB130:GOSUB135:GOSUB140
1003 PRINT"▒▒▒▒▒YOUR TIME:",TI$
1004 PRINT"▒▒▒▒▒YOUR SCORE:",SC
1005 PRINT"▒▒          ▒BYE BYE ...!▒▒":GOTO30000
10000 POKEP,10:GOSUB130:GOSUB135:GOSUB122
10001 PRINT"▒▒▒▒▒▒▒ WER NICHT HOEREN        WILL, MUSS FUEHLEN!":GOSUB122
10002 PRINT"▒▒▒▒▒ SIE HABEN SOEBEN ALLE IHRE▒LEBEN VERLOREN !":GOSUB122
10003 PRINT"▒▒▒▒▒ BEVOR SIE'S NOCHMAL   PROBIEREN, RELAXEN    SIE NOCH EIN WENIG
!!"
10004 FORHJ=1TO5000:NEXTHJ:POKEP-2,0:POKEP-3,0:POKE52,28:RUN
20000 POKEP,234:POKE36869,255:GOSUB130:GOSUB140
20002 PRINT"▒▒▒▒  MNO▒-SOFT WUENSCHT    IHNEN VIEL SP▒▒▒SS &      VIELE PUNKTE !
!"
20003 POKEP-1,15:FORGB=130TO254:POKEP-3,GB:POKEP-2,GB:FORNK=1TO40:NEXT:NEXT:POKE
P-1,0:POKEP-3,0:POKEP-2,0
20004 POKEP,8:GOTO5
30000 POKEP-2,0
30001 POKEP-1,15:FORBB=1TO33:READT:POKEP-3,T:FORSD=1TO40:NEXTSD:NEXTBB
30002 PRINT"▒▒":END
30010 DATA225,225,0,225,0,225,0,225,225,0,225,225,233,233,225
30020 DATA225,233,233,237,237,240,240,240,240,240,0,225,195,135,135,0,0,0

READY.
```

Space-Fire

für ZX 81

16 K RAM

Sie müssen, um zu überleben, die feindlichen Schiffe, welche versuchen, auf ihren Planeten zu landen, mit Ihrer Phaserkanone aufhalten.

Gelingt Ihnen das nicht, so verlieren Sie einen Energiepunkt. Haben Sie keine Energiepunkte, so verlieren Sie eines Ihrer drei eigenen Schiffe. Sind alle Ihrer drei Schiffe zerstört, so ist das Spiel beendet. Für die feindlichen Schiffe bekommen Sie je nach Art 10–100 Punkte. Der High-score liegt bei 300 Punkten. Überbieten Sie den High-score, so können Sie sich als High-score-Halter eintragen.

Ab und zu fliegt ein Versorgungsschiff vorbei. Treffen Sie es, so können Sie 100 weitere Punkte auf Ihren Scorestand buchen. Gelenkt wird Ihr Schiff nach rechts mit 0, mit 1 nach links. Feuern können Sie mit den Tasten 2 und 9. Ist Ihnen der High-score zu hoch bzw. zu niedrig, so können Sie diesen in Zeile 3 verändern.

Das Spiel wird nach dem Laden, welches mit Load"" möglich ist, sofort gestartet. Dann erscheint das Titelbild, welches Sie durch einen Tastendruck wegscrollen können.

```
   1 REM (C)O.BAETZ
            MAIGLOECKCHENST.20
            2964 WIESMOOR
            TELE 04944/3739
   2 LET A$="O.BAETZ"
   3 LET HS=300
   4 LET WO=7
   5 REM >>S P A C E-F I R E<<
   6 FOR L=0 TO 21
   7 SCROLL
   8 NEXT L
  10 PRINT AT 0,0;"
ICE      SPF
"
  11 IF PEEK 16517<>52 THEN RAND
USR 16514
  20 PRINT AT 10,10;"
E          FIR
"
  30 PRINT AT 8,24;""
  40 PRINT AT 10,2;"(C) BY";AT 1
2,2;"O.BAETZ"
  50 FOR L=0 TO 200
  51 IF INKEY$<>"" THEN GOTO 60
  55 NEXT L
  56 CLS
  60 FOR L=0 TO 21
  61 SCROLL
  65 NEXT L
  70 PRINT AT 0,0;"        S P A C
E - F I R E"
  80 PRINT AT 7,0;"  ..........50
POINTS             ..........10
POINTS             ..........40
POINTS             ..........30
POINTS"
  83 PRINT
  84 PRINT " FIRE..........<=2 9=
>          CONTROL.......<=1 0=
>"
  85 PRINT
  86 PRINT "  ........100 POINT
S"
  90 FOR L=0 TO 500
  91 IF INKEY$<>"" THEN GOTO 95
  92 NEXT L
  95 CLS
  99 FOR L=0 TO 21
 100 SCROLL
 103 NEXT L
 104 PRINT AT 0,0;
 110 FOR A=0 TO 21
 120 PRINT "
"
 130 NEXT A
 140 PRINT AT 1,0;" SCORE:      H-
S      ENERGIE:   AAA"
 150 PRINT AT 2,0;"
"
 165 LET C=10
 170 LET SC=0
 175 LET SS=C
 180 LET E=4
 190 LET D=28
 200 PRINT AT 1,7;SC;AT 1,15;HS;
AT 1,27;E
 290 LET M$=""
 299 REM HAUPTTEIL
 300 LET B=INT (RND*25)+2
 301 LET PL=INT (RND*50)+1
 308 LET M$=""
 309 LET DDT=INT (RND*2)+3
 310 FOR A=WO TO 21
 314 LET PP=INT (RND*3)
 315 IF PP=0 THEN LET B=B+1
 316 IF PP=30 THEN LET PP=29
 319 IF PP=2 THEN LET B=B-1
 320 IF DDT=4 THEN LET M$=""
 321 IF PP=2 THEN PRINT AT A,B;M
```

```
$;AT A-1,B+1;"█"
 322 IF PP=1 THEN PRINT AT A,B;M
$;AT A-1,B;"█"
 323 IF PP=0 THEN PRINT AT A,B;M
$;AT A-1,B-1;"█"
 324 IF DDT=3 THEN LET M$="▒"
 325 IF DDT=2 THEN LET M$="▒"
 330 IF INKEY$="0" THEN LET C=C+
1
 340 IF INKEY$="1" THEN LET C=C-
1
 341 IF FL=10 THEN FOR T=0 TO 27
 342 IF FL=10 THEN IF INKEY$="0"
THEN LET C=C+1
 343 IF FL=10 THEN IF INKEY$="1"
THEN LET C=C-1
 344 IF FL=10 THEN PRINT AT 21,C
-1;"█A█"
 345 IF FL=10 THEN PRINT AT 10,T
;"█+█"
 346 IF FL=10 THEN IF INKEY$="9"
OR INKEY$="2" THEN GOTO 2000
 347 IF FL=10 THEN NEXT T
 348 IF FL=10 THEN PRINT AT 10,2
7;"█████"
 350 IF C=1 THEN LET C=2
 360 IF C=27 THEN LET C=26
 370 PRINT AT 21,C-1;"█A█"
 380 IF INKEY$="2" OR INKEY$="9"
THEN GOTO 500
 390 IF A=21 THEN GOTO 500
 450 NEXT A
 451 LET DDT=0
 455 LET M$="▒"
 490 GOTO 200
 500 IF A=21 THEN LET E=E-1
 501 IF A=21 AND C=B THEN LET E=
0
 502 IF B=55 AND DDT=4 THEN LET
SC=SC+40
 503 IF B=55 AND DDT=3 THEN LET
SC=SC+30
 504 IF B=55 AND DDT=2 THEN LET
SC=SC+20
 509 IF B=55 THEN LET E=E+1
 510 IF E=0 THEN LET D=D+1
 515 IF D=32 THEN GOTO 1000
 519 IF E=0 THEN PRINT AT 1,D;"█
"
 520 IF E=0 THEN FOR Z=0 TO 2
 521 IF E=0 THEN PRINT AT 21,C-3
;"█ █";AT 20,C-2;"█ █";AT 21,C
-1;"█";AT 20,C-5;"█ █";AT 20,C;
"███";AT 21,C-3;"█████";AT 20,
C-2;"███";
 522 IF E=0 THEN PRINT AT 21,C-1
;"█";AT 20,C-5;"█████";AT 20,C;"
█████"
 523 IF E=0 THEN PRINT AT 18,C-3
;"▒";AT 17,C-2;"▒";AT 18,C-1;"▒"
;AT 17,C;"▒";AT 18,C-3;"█";AT 17
,C-2;"█";AT 18,C-1;"█";AT 17,C;"
█"
 529 IF E=0 THEN NEXT Z
 530 IF E=0 THEN LET E=3
 540 IF B<>55 THEN PRINT AT 1,27
;E
 545 LET 55=C
 550 IF A=21 THEN GOTO 299
 560 FOR S=19 TO A-3 STEP -2
 570 PRINT AT S,55;"█";AT S,55;"
█"
 571 IF INKEY$="0" THEN LET C=C+
1
 572 IF INKEY$="1" THEN LET C=C-
1
 573 IF INKEY$="0" OR INKEY$="1"
THEN PRINT AT 21,C-1;"█A█"
 580 NEXT S
 581 LET S=S+3
 590 IF B=55 THEN PRINT AT S,55;
"█ ";AT S+1,55-1;"▒▒▒";AT S+2,55
-1;"▒▒";AT S-1,55;"▒▒";AT S-3,S
S-2;"O M";AT S-2,55-3;"B O";AT S
-2,55;"██";AT S+1,55-1;"██";AT S+
2,55-1;"██";AT S-1,55;"████";AT S
-3,55-2;"███";AT S-2,55-3;"███"
 595 LET S=S-4
 600 IF B=55 THEN LET SC=SC+10
 603 IF B=55 AND DDT=4 THEN LET
SC=SC+50
 610 IF B=55 THEN PRINT AT 1,7;S
C
 630 IF SC=HS THEN PRINT AT 10,1
0;"HIGH-SCORE";AT 10,10;"HIGH-SC
ORE";AT 10,10;"██████████"
 650 IF SC>=HS THEN LET HS=SC
 660 IF SC>=HS THEN PRINT AT 1,1
5;SC
 700 IF B=55 THEN GOTO 299
 710 GOTO 381
1000 FOR A=0 TO 20
1010 PRINT AT 10,10;"GAME-OVER";
AT 10,10;"GAME-OVER"
1020 NEXT A
1021 IF SC>=HS THEN FOR O=0 TO 2
0
1022 IF SC>=HS THEN PRINT AT 8,0
;"HIGH-SCORE"
1023 IF SC>=HS THEN NEXT O
1030 PRINT AT 14,0;"DU HAST ";SC
;" PUNKTE "
1040 IF SC>=HS THEN PRINT "UND D
AMIT DEN HIGH-SCORE.       HERZL
ICHEN GLUECKWUNSCH.BITTE TRAGE D
EINEN NAMEN EIN."
1050 IF SC>=HS THEN INPUT A$
1060 PRINT "DER HIGH-SCORE IST V
ON ";A$
1070 PRINT "NOCH EINMAL? DANN DRU
ECKE "1""
1080 IF INKEY$="" THEN GOTO 1080
1090 CLS
1095 PAUSE 100
1100 GOTO 100
1110 STOP
1120 SAVE "SPACE-FIR▒"
1130 GOTO 4
1999 STOP
2001 FOR U=19 TO 10 STEP -1
2005 PRINT AT 10,T;"█ █"
2006 LET T=T+1
2010 PRINT AT U,C;"▌";AT U+1,C;"
█"
2020 NEXT U
2030 PRINT AT U,C;"█"
2040 IF C<>T+2 THEN GOTO 350
2045 PRINT AT U,C;"█"
2050 LET SC=SC+100
2060 PRINT AT 10,T;"▒▒▒▒▒";AT 9,
T;"██ OOB";AT 10,T;"█████";AT
9,T;"██████"
2100 GOTO 200
```

Korrektur

Concentration
für Commodore 64
Seite 20

Ausgabe 1/85
Bitte überprüfen Sie
nur die letzten Ziffern!

```
110 DATA133,254,169,208,133,255,162,0,160,0,177,254,145,252,200,208,249,232
120 DATA224,16,240,7,230,253,230,255,76,29,192,162,55,134,1,162,1,142,14
130 DATA220,169,0,133,252,169,160,133,253,162,0,160,0,177,252,145,252,200
140 DATA208,249,232,224,32,240,5,230,253,76,69,192,198,1,162,83,142,190,160
```

Ritter Kunibert

für Zx-Spectrum

Ritter Kuniberts Kampf gegen Burg Steinbrech

```
1 REM KUNIBERT
2 GO SUB 9000
10 LET hsc=0
20 LET sc=0: LET o=2
25 LET zae=0: INPUT "Schwierig
keitsgrad? 1-4 ";sch: LET sch=sc
h/10+.3: IF sch>.7 THEN GO TO 25
30 DIM a(6): DIM b(6)
90 BORDER 1: PAPER 7: INK 0: B
RIGHT 1: CLS
100 PRINT AT 6,0;"
110 PRINT AT 7,0;"
120 PRINT AT 8,0;"
130 PRINT AT 9,0;"
140 PRINT AT 10,0;"
150 PRINT AT 11,0;"
160 PRINT AT 12,0;"
170 PRINT AT 13,0;"
180 PRINT AT 14,0;"
190 FOR i=15 TO 21
200 PRINT "
210 NEXT i
215 LET li=1: PRINT AT 0,11; IN
K 1;"HSC=";hsc
220 FOR i=7 TO 23 STEP 3
230 FOR l=21 TO 7 STEP -1
240 PRINT AT l,i; INK 2; BRIGHT
1; PAPER 0;""
250 NEXT l: LET a(li)=i: LET li
=li+1: NEXT i
260 PRINT AT 5,24; INK 1; BRIGH
T 1;""
300 LET c=8: LET st=0
305 BRIGHT 1: PRINT AT 5,c; INK
1;"";AT 4,c; INK 1;""
310 DIM a$(4,3): LET a$(1)="
"
320 LET a$(2)="  "
330 LET a$(3)="  "
340 LET a$(4)="  "
500 LET x=INT (RND*6)+1
510 PRINT AT 21-b(x),a(x); INK
2; BRIGHT 1; PAPER 0;""
520 LET b(x)=b(x)+sch: IF INT (
b(x)+.5)=15 THEN GO TO 2000
530 PRINT AT 21-b(x),a(x); INK
4; PAPER 0;""
550 IF INKEY$<>"" THEN GO SUB 7
00
560 GO TO 500
700 IF INKEY$="x" THEN GO TO 80
0
710 IF INKEY$="z" THEN GO TO 80
0
730 IF INKEY$="m" AND st=1 AND
o=1 THEN GO TO 800
740 IF INKEY$="m" AND st=1 AND
o=2 THEN GO TO 900
800 IF INKEY$="m" AND st=1 THEN
GO SUB 1000
805 LET c1=c
810 LET c=c+(INKEY$="x" AND c<2
4)
815 LET o=1
820 IF c=24 THEN LET st=1
830 PRINT AT 5,c1;"  ";AT 4,c1-1
;"   "
840 IF st=0 THEN PRINT AT 4,c-1
;a$(3)
850 IF st=1 THEN PRINT AT 4,c-1
;a$(2)
855 PRINT AT 5,c;""
860 RETURN
900 IF INKEY$="m" AND st=1 THEN
GO SUB 1500
```

RITTER KUNIBERT, Herr und Gebieter der „Burg Steinbrech", ist der letzte lebende Nachkomme einer heldenhaften Ahnenfolge derer von und zu „Steinbrech", die schon seit ewigen Zeiten mit den Nachkommen derer von „Steinbrecher" in Blutsfehde leben.

Nach einem erneuten Streit beider Familien hat Ritter Steinbrecher seinen Soldaten den Befehl zur Erstürmung der „Burg Steinbrech" gegeben. Sie allein haben durch Ritter Kunibert die Möglichkeit, die angreifenden Soldaten auf der Westseite der Burg in einen erbitterten Kampf zu verwickeln. Ritter Kunibert kann sich gegen die Angreifer durch Steinwürfe zur Wehr setzen. Dabei muß er darauf achten, daß die angreifenden Soldaten mit ihren Leitern nicht in das Innere der Burg gelangen können. In diesem erbitterten und erfolglos scheinenden Kampf ist es also seine Aufgabe, soviel Angreifer wie möglich durch herabgeworfene Steine am Eindringen in seine Burg zu hindern. Jeder getroffene Soldat oder jede getroffene Leiter erhöht die Punktzahl. Helfen Sie also Ritter Kunibert, die Eroberung seiner „BURG STEINBRECH" durch die feindlichen Angreifer zu verhindern.

Steuerung:
Z = links
X = rechts
M = werfen
Start = beliebige Taste

Die Steuerung erfolgt über Tastatur.

* * * * * *

```
 905 LET c1=c
 910 LET c=c-(INKEY$="z" AND c>7
)

 915 LET o=2
 920 PRINT AT 5,c1;" ";AT 4,c1-1
;"  "
 930 IF st=0 THEN PRINT AT 4,c-1
,a$(4)
 940 IF st=1 THEN PRINT AT 4,c-1
;a$(1)
 950 PRINT AT 5,c;"Ⓐ"
 960 RETURN
1000 LET d=c+1
1010 FOR i=1 TO 6: IF a(i)=d THE
N GO TO 1600
1020 NEXT i: RETURN
1500 LET d=c-1
1510 FOR i=1 TO 6: IF a(i)=d THE
N GO TO 1600
1520 NEXT i: RETURN
1600 LET b(i)=0
1605 LET s1=0: PRINT AT 4,d;" ":
 FOR i=7 TO 21
1610 PRINT AT i-1,d; INK 2; PAPE
R 0;"Ⓑ";AT i,d; INK 6;"◆"

1615 BEEP .008,i
1620 NEXT i: PRINT AT 21,d; INK
2;"Ⓑ"
1630 LET sc=sc+10
1640 PRINT #0;AT 1,11; INK 6;"SC
ORE=";sc
1645 LET zae=zae+1: IF zae=20 TH
EN LET zae=0: LET sch=sch+.1
1650 RETURN
2000 LET g=a(x): IF c>g THEN LET
 x=1
2010 IF c<g THEN LET x=-1
2020 IF c=g THEN GO TO 2100
2030 BEEP .01,-4: BEEP .01,4: BE
EP .01,20
2040 LET g1=g: LET g=g+x
2050 PRINT AT 5,g1;" ";AT 5,g; I
NK 4; BRIGHT 1;"Ⓧ"
2060 IF c=g THEN GO TO 2100
2070 GO TO 2030
2100 PRINT AT 4,c;" "
2110 FOR i=7 TO 20

2120 PRINT AT i-1,c;"■";AT i,c;
INK 6; PAPER 0;"Ⓒ";AT i+1,c; INK
 6; PAPER 0;"Ⓐ"
2130 BEEP .02,-20: NEXT i
2140 FOR i=1 TO 10: PRINT AT 5,g
;" ": PRINT AT 4,g; INK 4;"Ⓧ": B
EEP .1,20
2150 PRINT AT 4,g;" ";AT 5,g; IN
K 4;"Ⓧ": BEEP .1,-10: NEXT i
2200 IF sc>hsc THEN LET hsc=sc:
BEEP 1,40
2210 CLS : PRINT AT 11,0;"Du Has
t ";sc;" Punkte erreicht"
2220 PRINT '''"Neues Spiel?  J/n"
2230 IF INKEY$="" THEN GO TO 223
0
2240 IF INKEY$="J" THEN GO TO 20
2250 PAPER 0: CLS : FOR i=1 TO 6
: FOR l=1 TO 7
2260 PRINT AT 11,9; INK l;"TSCUE
SS"
2270 BEEP .01,i*l

2280 NEXT l: NEXT i
2285 PAPER 7: INK 0: CLS
2290 STOP
9000 FOR i=1 TO 7: READ a$: FOR
l=0 TO 7: READ a: POKE USR a$+l,
a: NEXT l: NEXT i
9100 DATA "a",24,153,90,60,24,60
,66,195
9110 DATA "b",24,60,126,255,255,
126,60,24
9120 DATA "c",195,255,255,195,19
5,255,255,195
9130 DATA "d",112,216,240,225,11
3,58,124,124
9140 DATA "e",124,124,124,68,68,
68,68,51
9150 DATA "f",14,27,15,135,142,9
2,62,62
9160 DATA "g",62,62,62,34,34,34,
34,204
9200 BORDER 2: PAPER 7: INK 0: C
LS

9210 PRINT AT 0,8;"KUNIBERT"
9220 PRINT "Ritter Kuniberts Bur
g wird      angegriffen"
9230 PRINT "Die einzige Moeglich
keit zur     Verteidigung ist auf
 der RechtenSeite der Burg Stein
e zu Hohlen und sie auf die Angr
eifer auf    den Sturmleitern zu
Werfen"
9240 PRINT "Erreicht ein Angreif
er den      Oberen Rand der Burg
 so hast du Verlohren"
9250 PRINT "Gesteuert wird Kunib
ert mit:     z=Links
             x=Rechts"
9260 PRINT "und m=Werfen"
9270 PRINT '''"Zum Starten eine T
aste druecken"
9280 IF INKEY$="" THEN GO TO 928
0
9290 FOR i=1 TO 15: BEEP .01,-10
: BEEP .01,-8: BEEP .01,-6: BEEP
 .01,-4: NEXT i: RETURN
```



U-Jäger
für Zx-Spectrum

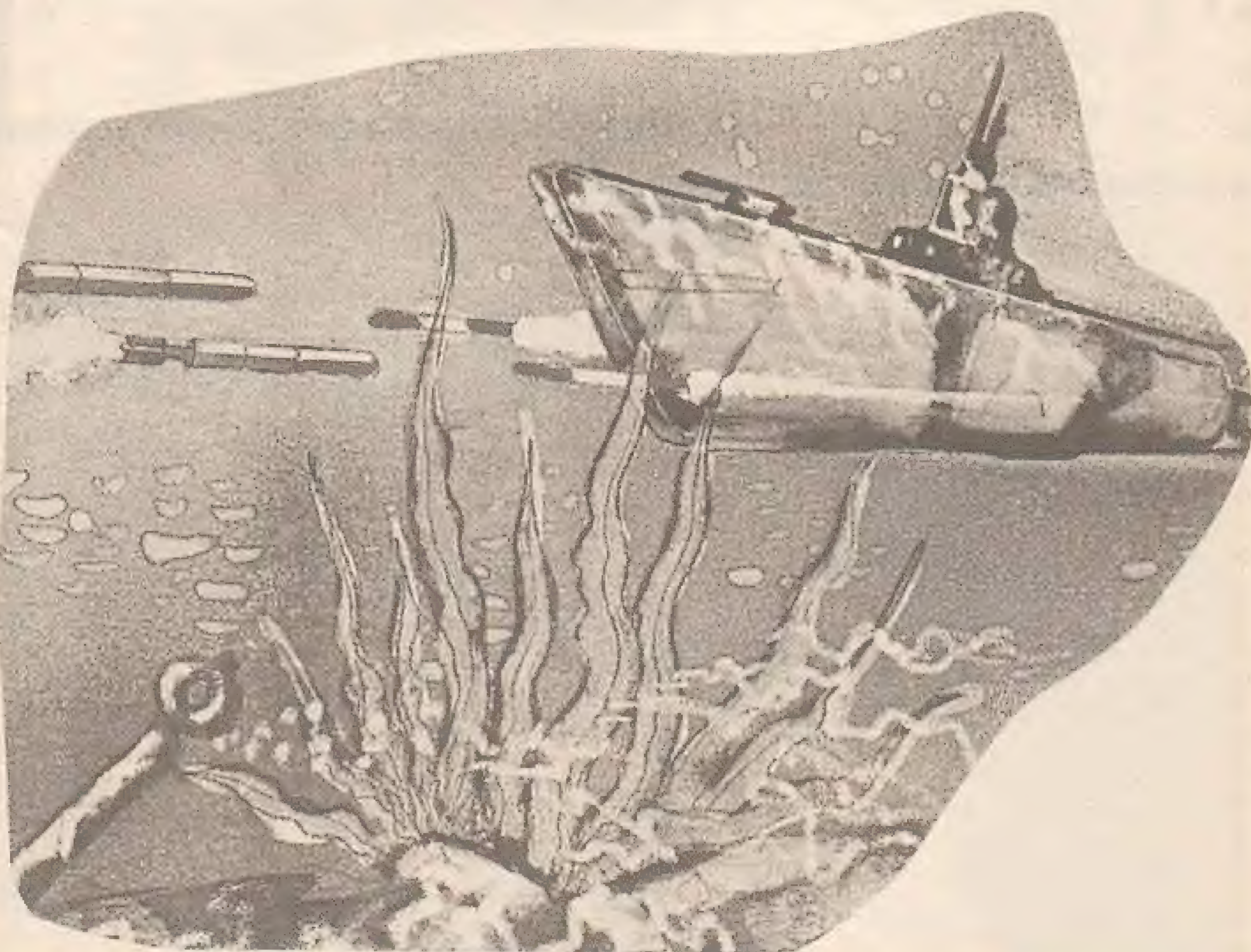
Das Programm U-JAEGER ist größtenteils in Maschinensprache geschrieben und ist daher sehr schnell.

Bei der Eingabe des Spiels sollte erst das extra beigefügte BASIC-Programm eingegeben werden, und dann damit die Dezimalwerte des Maschinenprogramms. Danach wird der eigentliche BASIC-Teil des Programms eingegeben.

Die REM-Zeilen des Programms können weggelassen werden, aber wir empfehlen keine weiteren Änderungen des Programms, da es sich sonst „aufhängen" könnte.

Im Spiel selbst geht es darum, daß man mit einem U-Jäger innerhalb eines bestimmten Zeitlimits möglichst viele U-Boote versenkt, ohne selbst getroffen zu werden. Ein versenktes U-Boot bringt Pluspunkte, während ein U-Boot, dem es gelingt, den Bildschirm zu durchqueren, Punkte kostet. Je tiefer ein U-Boot taucht, desto mehr Punkte bringt oder kostet es. Wird man selbst versenkt, so bedeutet es nicht unbedingt das Ende, aber es kostet ziemlich viel Punkte. Und von den Punkten ist abhängig, ob man noch mehr Zeit bewilligt bekommt oder nicht.

Das Schiff wird mit der linken untersten Tastenreihe (CAPS SHIFT bis V) nach links und mit der Tastenreihe von B bis SPACE nach rechts bewegt. Mit der Reihe von A bis G werden links vom Schiff Wasserbomben abgeworfen und mit H bis ENTER rechts.



```
  10 REM ***********************
  20 REM *                     *
  30 REM *      U - JAEGER     *
  40 REM *                     *
  50 REM *    (C)  1984  by    *
  60 REM *  ALEXANDER  WALTER  *
  70 REM *    FONTANESTR. 7    *
  80 REM *    8900 AUGSBURG    *
  90 REM *                     *
 100 REM ***********************
 105
 110 GO TO 1000
 120 PRINT #1: RANDOMIZE USR 31138
 130 PRINT #1;AT 1,0; PAPER 4; INK 0;"TIME:      *****  POINTS:      "
 140 RANDOMIZE USR 31356
 150 IF PEEK 23305=128 THEN  BEEP .01,1: BEEP .01,10: BEEP .01,-5: GO TO 140
 160 IF PEEK 23305=1 THEN  GO TO 200
 170 FOR x=1 TO 30: NEXT x
 180 GO TO 140
 185
 190 REM *** ENDE ***
 200 LET p=PEEK 23302*10+PEEK 23303*2560
 210 FLASH 1: PAPER 2: INK 7
 230 IF H>=P THEN  GO TO 400
 300 INPUT "Gratuliere, sie haben einen     neuen HS erreicht. Geben sie    ihre
n Namen ein (max. 9 B.)", LINE H$
 310 IF LEN H$>9 OR H$="" THEN  GO TO 300
 320 FOR X=1 TO LEN H$
 330 IF H$(X)<" " OR H$(X)>"ø" THEN  GO TO 300
 340 NEXT X
 360 PRINT #1;AT 0,0;"    NEUER HIGHSCORE : ";P,
 370 PRINT #1;"    ALTER HIGHSCORE : ";H,
 380 LET h=p
 390 PRINT
 400 PRINT AT 20,0;" Willst du nocheinmal spielen ? "
 410 PRINT TAB 20;"( Y / N )",
 420 PRINT AT 0,0; FLASH 0; PAPER 0;" Highsc.: ";H;TAB 17;"von: ";H$,
 430 LET Z$=INKEY$: IF Z$="" THEN  GO TO 430
 440 IF Z$="Y" OR Z$="y" THEN  GO TO 120
 450 IF Z$<>"N" AND Z$<>"n" THEN  GO TO 430
 460 GO TO 1e4
 990
1000 REM *** ANFANG ***
1010 BORDER 0: LET h=0: LET h$=""
1020 GO SUB 3000
1030 PRINT '"Willst du eine Spielerklaerung ?"'TAB 24;"(Y/N)"
1040 LET Z$=INKEY$: IF Z$="" THEN  GO TO 1040
1050 IF Z$="Y" OR Z$="y" THEN  GO TO 1200
1060 IF Z$<>"N" AND Z$<>"n" THEN  GO TO 1040
1190 GO TO 120
1200 GO SUB 3000
1205 PRINT
1210 PRINT "Du bist Commandant eines"
1220 PRINT "U-Jaegers und musst moeglichst"
1230 PRINT "viele U-Boote versenken, wobei"
1240 PRINT "Dir aber ein Zeitlimit gesetzt"
1250 PRINT "ist."
1255 PRINT
1260 PRINT "Nur wenn Du viele Punkte er-"
1270 PRINT "reichst wird Dir Bonuszeit"
1280 PRINT "gewaehrt."
1285 PRINT
```

```
1290 PRINT "Jedes U-Boot,dass nicht versenkt"
1300 PRINT "wird, kostet Punkte, genauso,"
1310 PRINT "wenn du selber versenkt wirst."
1320 PAUSE 0: GO SUB 3000
1330 PRINT
1340 PRINT "Mit CAPS SHIFT bis V steuert man"
1350 PRINT "nach links und mit B bis SPACE"
1360 PRINT "nach rechts."
1370 PRINT
1380 PRINT "Mit A bis G wirft man die Bomben"
1390 PRINT "links vom Schiff ab und mit"
1400 PRINT "H bis ENTER wirft man rechts ab."
1410 PAUSE 0
2000 GO TO 120
2990
3000 FLASH 0: PAPER 0: INK 7: CLS
3010 PRINT "**********************************";TAB 31;"*"
3020 PRINT "*";TAB 11;"U - JAEGER";TAB 31;"*"
3030 PRINT "*";TAB 31;"*"
3040 PRINT "*  © 1984 BY ALEXANDER WALTER  *"
3050 PRINT "*";TAB 31;"**********************************"
3060 PRINT : RETURN
7990
7999 STOP
8000 CLEAR 31000: LOAD ""CODE : RUN
8500
9000 SAVE "u-jaeger" LINE 8000
9010 SAVE "u-jaeger"CODE 31001,1729
9020 VERIFY "": VERIFY ""CODE
9999 BORDER 7: INK 0: PAPER 7: FLASH 0: BRIGHT 0
```

```
3000>REM  Eingaberoutine fuer das
3010 REM  Maschienmnprogramm  "U - JAEGER"
3020 REM
3030 REM
3040 REM  Am besten erst dieses Programm
3050 REM  eingeben und dann damit die Werte einpoken.
3070 REM
3080 REM  Nach dem Starten wird die Speicherstelle
3090 REM  und der bisherige Wert angegeben.
3095 REM
3100 REM  Wenn nur ENTER eingegeben wird,
3110 REM  wird der alte Wert belassen.
3115 REM
3120 REM  Mit einer Zahl kleiner  als 0
3130 REM  geht man um eine Speicherstelle zurueck,
3140 REM  um sie zu korigieren.
3150 REM
3160 REM  Und mit 10000 wird abgespeichert.
3170 REM
3180 REM  Wenn die Kontrollwerte nicht stimmen
3190 REM  dann hast du einen Tippfehler gemacht !
3200
3210
4000 CLEAR 31000: LET a=31001: REM   *** DAS CLEAR IST WICHTIG ***
4040 LET C1=A: LET C2=A: LET C3=A: LET X=-1
4050 LET X=X+1
4060 LET C=A+X: INPUT (C;"(";PEEK C);") ";z$: IF z$="" THEN  LET z$=STR$ PEEK c
4070 LET Z=INT VAL Z$: IF Z=10000 THEN  GO TO 5000
4080 IF Z>255 THEN  GO TO 4060
4090 IF Z<0 AND X>0 THEN  LET X=X-1: LET Z=PEEK (A+X): LET C1=C1-Z: LET C2=C2-Z+
2*Z*(C<>2*INT (C/2)): LET C3=C3-Z+2*Z*(C=2*(INT C/2)): GO TO 4060
```

```
4100 POKE C,Z: PRINT C;" - ";Z
4200 LET C1=C1+Z: LET C2=C2+Z-2*Z*(C<>2*INT (C/2)): LET C3=C3+Z-2*Z*(C=2*(INT C/
2))
4210 IF X<99 THEN  GO TO 4050
4300 PRINT "SUMME : ";C1'"1. DIVERENZ : ";C2'"2. DIVERENZ : ";C3''
4400 INPUT "ALLES IN ORDNING ? (Y/N) ";A$
4410 IF A$="N" OR A$="n" THEN  GO TO 4060
4420 IF A$<>"Y" AND A$<>"y" THEN  GO TO 4400
4430 LET A=A+100: GO TO 4040
4500
4600 LOAD ""CODE : GO TO 4060
5000 PRINT "*********  WICHTIG  *********"
5010 PRINT "  NOCH ""VOR"" DEM NEULADEN "
5020 PRINT "  ***  CLEAR 31000 *** EINGEBEN"
5030 PRINT "SONST LERNST DU UNTER UMSTAENDEN DAS FLUCHEN"
5050 SAVE "u - j" LINE 4600
5060 SAVE "u-jaeger"CODE 30978,1790
```

U - JAEGER

31001	106	38	0	41	41	41	41	41	6	88
31011	75	9	126	201	245	62	22	215	122	215
31021	123	215	241	215	201	197	6	8	205	64
31031	121	31	203	25	16	248	121	193	201	197
31041	58	120	92	79	58	1	91	230	31	71
31051	10	169	50	1	91	230	1	193	201	22
31061	0	114	35	11	120	177	200	24	248	229
31071	33	150	125	213	87	62	255	190	35	32
31081	252	21	32	249	21	126	35	186	40	3
31091	215	24	248	209	225	201	6	16	175	87

SUMME : 42077 1. DIVERENZ : 31493 2.DIVERENZ : 19925

31101	95	41	23	203	35	203	18	28	145	48
31111	2	129	29	16	242	201	175	245	14	10
31121	205	121	121	198	81	245	235	124	181	32
31131	243	241	167	200	215	24	250	33	0	126
31141	37	34	54	92	33	0	64	1	0	24
31151	205	84	121	14	64	22	45	205	86	121
31161	1	192	2	22	9	205	86	121	1	250
31171	0	205	84	121	62	12	50	143	92	6
31181	32	17	1	0	62	8	205	39	121	120
31191	230	3	198	36	215	16	248	62	5	50

SUMME : 40683 1. DIVERENZ : 30905 2.DIVERENZ : 21519

31201	2	91	33	0	0	34	120	92	33	86
31211	19	34	4	91	205	66	22	30	10	195
31221	252	122	197	205	25	121	193	254	15	192
31231	33	14	91	122	17	4	0	43	43	25
31241	190	32	252	209	43	43	54	1	62	128
31251	50	9	91	33	2	91	52	221	54	0
31261	0	195	110	123	237	120	230	63	254	63
31271	200	33	2	91	175	190	200	53	33	64
31281	91	43	35	35	190	32	251	22	1	114
31291	43	115	62	41	195	39	121	205	50	121

SUMME : 40195 1. DIVERENZ : 30907 2.DIVERENZ : 22207

31301	230	7	254	6	210	205	124	135	135	87
31311	30	0	221	33	14	91	221	25	221	126
31321	0	254	3	194	205	124	221	126	2	119
31331	43	221	126	3	198	3	119	35	195	205
31341	124	42	6	91	237	82	48	3	33	0
31351	0	34	6	91	201	205	66	22	33	0

```
31361   126   37    34    54    92    33    9     91    54    0
31371   58    8     91    167   202   217   122   54    128   17
31381   12    0     205   110   122   61    50    8     91    40
31391   50    61    40    35    22    0     58    0     91    95
SUMME : 40141       1. DIVERENZ : 29775               2.DIVERENZ : 22461

31401   62    40    50    143   92    62    32    205   39    121
31411   62    3     205   94    121   20    62    32    205   39
31421   121   62    1     205   94    121   195   39    123   62
31431   45    50    143   92    62    11    205   94    121   24
31441   85    33    0     91    94    53    24    35    58    0
31451   91    95    1     254   254   237   120   230   63    254
31461   63    40    3     29    40    60    1     254   127   237
31471   120   230   63    254   63    40    6     28    62    24
31481   187   40    43    58    0     91    187   40    37    123
31491   50    0     91    62    40    50    143   92    22    0
SUMME : 40132       1. DIVERENZ : 31776               2.DIVERENZ : 22670

     U - JAEGER

31501   62    32    205   39    121   62    1     205   94    121
31511   62    32    20    205   39    121   205   25    121   54
31521   45    62    2     205   94    121   6     5     221   33
31531   64    91    62    14    50    143   92    221   94    0
31541   221   35    221   86    0     175   186   40    48    205
31551   247   121   254   40    40    24    62    1     186   54
31561   9     32    2     54    45    62    21    186   32    10
31571   33    2     91    52    221   54    0     0     24    17
31581   221   52    0     20    205   247   121   205   50    121
31591   230   3     198   40    205   39    121   221   35    16
SUMME : 40495       1. DIVERENZ : 30517               2.DIVERENZ : 22507

31601   192   58    8     91    167   32    27    62    41    50
31611   143   92    1     254   253   58    0     91    95    205
31621   33    122   1     254   191   58    0     91    198   7
31631   95    205   33    122   221   33    14    91    33    3
31641   91    52    126   230   1     79    6     6     197   62
31651   9     50    143   92    221   94    3     221   86    2
31661   221   126   0     167   32    85    122   179   40    23
31671   195   196   123   213   122   135   22    0     95    205
31681   110   122   209   62    32    205   39    121   62    3
31691   205   94    121   221   54    3     0     205   64    121
SUMME : 41451       1. DIVERENZ : 32457               2.DIVERENZ : 21751

31701   32    6     62    255   221   54    3     25    221   119
31711   1     221   54    0     3     221   229   205   50    121
31721   230   15    198   4     221   33    14    91    17    4
31731   0     46    6     221   190   2     40    235   221   25
31741   45    32    246   221   225   193   221   119   2     24
31751   153   254   3     40    32    62    79    50    143   92
31761   62    32    205   39    121   62    4     205   94    121
31771   106   38    0     237   91    6     91    41    25    34
31781   6     91    221   53    0     24    54    175   221   70
31791   1     185   40    1     120   131   95    203   72    32
SUMME : 41262       1. DIVERENZ : 31680               2.DIVERENZ : 22140

31801   8     62    25    187   202   186   123   24    5     175
31811   187   202   186   123   62    32    205   39    121   62
31821   15    50    143   92    221   115   3     121   221   203
31831   1     78    40    2     198   2     198   5     205   94
31841   121   193   17    4     0     221   25    5     194   161
31851   123   33    44    91    6     6     94    35    86    175
31861   186   202   66    122   229   197   205   25    121   254
31871   74    32    2     54    9     254   40    32    5     62
```

```
31881   4     50    8     91    122   254   1     32    4     54
31891   45    24    52    62    74    50    143   92    21    21
SUMME : 41033       1. DIVERENZ : 32053             2.DIVERENZ : 22569

31901   32    30    20    62    42    50    143   92    205   25
31911   121   254   40    32    10    62    2     50    143   92
31921   62    4     50    8     91    62    34    205   39    121
31931   24    13    20    205   25    121   254   9     32    5
31941   62    33    205   39    121   193   225   53    35    16
31951   161   205   52    22    17    24    1     62    32    50
31961   143   92    205   39    121   42    6     91    205   141
31971   121   62    81    215   62    32    215   17    4     1
31981   62    32    205   39    121   42    4     91    237   91
31991   120   92    167   237   82    235   48    83    42    4
SUMME : 40329       1. DIVERENZ : 31227             2.DIVERENZ : 23473

    U - JAEGER

32001   91    17    184   11    25    34    4     91    42    10
32011   91    17    232   3     25    34    10    91    237   91
32021   6     91    237   82    48    113   33    0     60    34
32031   54    92    205   66    22    62    207   50    143   92
32041   62    9     205   94    121   42    120   92    6     150
32051   62    9     50    143   92    118   16    253   34    120
32061   92    62    10    205   94    121   205   52    22    33
32071   0     126   37    34    54    92    62    32    50    143
32081   92    235   14    50    205   121   121   235   205   141
32091   121   62    32    215   17    11    1     62    32    205
SUMME : 40539       1. DIVERENZ : 32159             2.DIVERENZ : 23463

32101   39    121   6     5     62    32    215   16    251   17
32111   12    1     62    8     205   39    121   58    2     91
32121   167   40    6     71    62    40    215   16    251   33
32131   0     60    34    54    92    1     0     0     201   62
32141   1     50    9     91    33    0     0     24    236   255
32151   44    45    46    47    48    49    32    255   50    51
32161   52    53    54    55    16    5     32    255   32    32
32171   32    32    32    32    255   76    77    78    79    80
32181   255   66    67    68    69    70    255   71    72    73
32191   74    75    255   56    57    58    59    60    255   61
SUMME : 39670       1. DIVERENZ : 30508             2.DIVERENZ : 24532

32201   62    63    64    65    16    1     32    255   22    21
32211   7     42    42    42    32    66    79    78    85    83
32221   90    69    73    84    32    42    42    42    255   22
32231   21    0     6     6     8     255   22    0     0     6
32241   6     6     6     255   0     51    59    108   100   32
32251   97    44    51    50    59    0     0     0     0     0
32261   0     0     0     0     126   90    126   126   90    126
32271   0     70    145   61    102   82    90    239   165   0
32281   0     0     0     0     0     0     0     0     6     15
32291   63    127   255   255   255   24    62    63    127   255
SUMME : 38502       1. DIVERENZ : 32542             2.DIVERENZ : 25900

32301   255   255   255   0     0     0     224   248   252   255
32311   255   0     48    56    124   126   255   255   255   0
32321   28    28    28    28    28    0     0     0     16    56
32331   124   62    28    8     0     0     0     24    60    60
32341   24    0     0     0     0     62    62    62    0     0
32351   0     0     0     0     0     0     0     3     31    0
32361   0     0     14    126   15    63    127   0     0     7
32371   31    28    255   255   234   96    96    224   248   248
32381   255   239   167   0     0     0     112   126   240   252
```

```
32391   254   0     0     0     0     0     0     192   248   255
SUMME : 40648       1. DIVERENZ : 31352             2.DIVERENZ : 23954

32401   121   57    31    15    7     3     1     255   255   153
32411   159   255   255   213   255   255   255   255   158   255
32421   255   85    255   255   255   255   121   255   255   85
32431   255   255   255   153   249   255   255   85    255   255
32441   158   156   248   240   224   192   128   0     0     255
32451   66    63    31    15    7     0     0     255   4     255
32461   255   255   255   0     1     255   3     255   255   255
32471   255   0     248   255   248   255   255   255   255   0
32481   0     224   32    240   240   224   192   0     0     15
32491   4     3     1     0     0     0     0     255   32    255
SUMME : 47791       1. DIVERENZ : 33183             2.DIVERENZ : 17011

    U - JAEGER

32501   255   255   127   0     0     255   128   255   255   255
32511   255   0     31    255   63    255   255   255   255   0
32521   0     254   2     255   255   254   252   0     0     7
32531   4     15    15    7     3     0     15    255   15    255
32541   255   255   255   0     128   255   192   255   255   255
32551   255   0     0     255   16    255   255   255   255   0
32561   0     255   66    252   248   240   224   0     0     127
32571   64    255   255   127   63    0     240   255   252   255
32581   255   255   255   0     0     255   1     255   255   255
32591   255   0     0     255   4     255   255   255   254   0
SUMME : 47921       1. DIVERENZ : 33927             2.DIVERENZ : 17081

32601   0     240   32    192   128   0     0     0     15    21
32611   37    44    33    19    28    227   20    93    71    121
32621   42    14    115   28    228   42    42    88    22    113
32631   143   112   67    28    164   161   159   196   124   0
32641   152   100   164   148   36    104   144   28    34    34
32651   34    34    34    28    0     8     24    40    8     8
32661   8     62    0     28    34    2     4     8     16    62
32671   0     28    34    2     12    2     34    28    0     8
32681   16    32    40    62    8     8     0     62    32    60
32691   34    2     34    28    0     12    16    32    60    34
SUMME : 37916       1. DIVERENZ : 32952             2.DIVERENZ : 27286

32701   34    28    0     62    2     4     8     8     8     8
32711   0     28    34    34    28    34    34    28    0     28
32721   34    34    30    2     34    28    0     126   255   0
32731   0     0     0     0     0     0     0     0     0     0
32741   0     0     0     0     0     0     0     0     0     0
32751   0     0     0     0     0     0     0     0     0     0
32761   0     0     0     0     0     0     0     0     0     0
32771   0     0     0     0     0     0     0     0     0     0
32781   0     0     0     0     0     0     0     0     0     0
32791   0     0     0     0     0     0     0     0     0     0
SUMME : 33654       1. DIVERENZ : 32652             2.DIVERENZ : 31748
```

*** U - JAEGER ***



So machen Sie mehr aus Ihrem COMMODORE 64:

Tips & Tricks





DER BESTSELLER – BAND 1

64 Tips & Tricks, das mit über 70.000 Exemplaren meistverkaufte DATA BECKER BUCH, ist eine hochinteressante Sammlung von Anregungen zur fortgeschrittenen Programmierung des COMMODORE 64, POKE's und andere nützliche Routinen, interessanten Programmen. Aus dem Inhalt: 3D-Graphik in BASIC – Farbige Balkengraphik – Definition eines eigenen Zeichensatzes – Tastaturbelegung – Simulation der Maus mit einem Joystick – BASIC für Fortgeschrittene – C-64 spricht deutsch – CP/M auf dem COMMODORE 64 – Druckeranschluß über den USER-Port – Datenübertragung von und zu anderen Rechnern – Synthesizer in Stereo – Retten einer nicht ordnungsgemäß geschlossenen Datei – Erzeugen einer BASIC-Zeile in BASIC – Kassettenpuffer als Datenspeicher – Multitasking auf dem COMMODORE 64-POKE's und die Zeropage.– GOTO, GOSUB und RESTORE mit berechneten Zeilennummern, INSTR und STRING-Funktion – Repeat-Funktion für alle Tasten. Alle Maschinenprogramme mit BASIC-Ladeprogrammen.

64 Tips & Tricks ist eine echte Fundgrube für jeden COMMODORE 64 Anwender. **64 TIPS & TRICKS, 1984, über 300 Seiten, DM 49.–**

JETZT NOCH MEHR TIPS & TRICKS – BAND 2

Auch der zweite Band von 64 Tips & Tricks dürfte sehr schnell ein Bestseller werden. Das Buch enthält eine Fülle hochkarätiger Programme, Anregungen und Routinen: ein umfangreiches Kapitel über Softwareschutz – Befehlserweiterungen und wie man sie macht – Tips & Tricks zur Programmierung von Superspielen – Zeiger und deren Manipulation – mehr übers Interrupt-Handling mit vielen Beispielen – erweiterte Hardware-Möglichkeiten – Betriebssystem ins RAM kopieren und dort manipulieren – sowie viele weitere Programme, Befehlserweiterungen und nützliche Routinen. Wer gerne programmiert und mehr wissen will über den COMMODORE 64, der braucht dieses neue Buch.
64 TIPS & TRICKS Band 2, ca. 250 Seiten, DM 39,–



DATA BECKER'S GROSSE PROGRAMM-SAMMLUNG ZUM COMMODORE 64, 250 Seiten, DM 49,–



DER COMMODORE 64 UND DER REST DER WELT, 220 Seiten, DM 49,–



DAS TRAININGSBUCH ZU SIMON's BASIC, 380 Seiten, DM 49,–



COMMODORE 64 FÜR TECHNIK UND WISSENSCHAFT, 300 Seiten, DM 49,–



DAS IDEENBUCH ZUM COMMODORE 64, 240 Seiten, DM 29,–

Diese und viele weitere DATA BECKER BÜCHER gibt's im Buchhandel, im Computerfachhandel und in den Warenhäusern. Dort gibt's auch den kostenlosen, großen DATA BECKER Katalog mit der großen Buch- und Softwareauswahl rund ums Thema Computer. Katalog auch kostenlos direkt von DATA BECKER.

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1

Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name und Adresse bitte deutlich schreiben

Kleinanzeigen

Verkäufe

TI-99/4A: TI+EX-Basic-programme verk. Info gegen Rückporto. B. Knedel, Tulpengasse 16, 3171 Weyhausen, Telefon 0 53 62 / 7 11 87

Dragon 32, suche Programme, z. B. Adventures, Simulations-Programme, Action-Games etc., Jun Kurokoshi-Krüger, Kattendahlerstr. 36, 4006 Erkrath

C-64: superschnelle Action: **3D-FORTRESS** (Raumjäger, Ufos, Energiescheiben, Laser etc.) und **3D-Vampir** und **PANZERFIGHT** auf Tape = **15,–** + 5,– DM Porto (Schein/V-Scheck). **Lieferung 48 Std.** C. Wurzer, Grüntenweg 14, 85 Nürnberg

M. Wolf, Tel. 06 41/4 14 50, tausche ca. 600 Programme

Sinclair ZX Spectrum 48 K bei uns nur DM 448,–, Sinclair QL DM 1798,–, Microdrive u. Interf. je DM 199,– bei Computer-Store, Herzebrocker Str. 46, 4830 Gütersloh, Tel. 0 52 41/1 20 80

Apple II: E. T. (Alleslerner) o. Geldspieler. Tamm, Solmsstr. 43, 1 Berlin 61

Verkaufe: Drucker Seikosha GP 500 A und SHARP PC 1260. Angebote an Chiffre ■

Verk. ZX 81 (Fol.-Tast. Def.) + 16K + Netzt. + Koaxkabel + Rec.-Kab. + deut. Anltg. + Buch M.80 Prgr., zus. 120,– DM. Chr. Lenk, Langlütjensand 4, 28 HB 66, Tel. 04 21 / 58 73 26

Verk. **Colour-Genie 32K,** mit original Software NW 600,–, 3 Handbüchern u. Datenrecorder für nur 550,–. Tel. 0 47 21 / 4 72 80

Verkaufe Super-Grafik-Abenteuer „Jetset-Freddy". Mit Highscoreliste, 2 Spielstufen und 6 Höhlen (auf Cassette)! 15,– DM an Bernd Schissler, Hölderlinstr. 3, 7117 Bitzfeld

Verk. TI 59 + Drucker PC100C + Anleitungen + Netzteil + 60 Magnetk. + 2 Rollen Papier, zus. für 450,– DM. W. Lenk, Langlütjensand 4, 28 HB 66, Tel. 04 21 / 58 73 26

FORTH-Assemblerlistings für alle wichtigen Mikroprozessoren je 25,– DM; Installation-Manual 15,– DM, Autolinker u. Buchhaltung f. Apple II. Kostenl. Info anf. bei B. Lippens – Datentechnik, Wilbrechtstr. 65, 8000 München 71

Verkaufe: TI-99/4A + Ext. Basic + Joysticks + Recorder + Kabel + TI-Invader + dt. Handbuch + ca. 70 PGM. + 1 TI-Magazin + Chip-Buch. VB: 700,– DM. Tel.: 02 21 / 60 76 76

Verkaufe C64 Spieleliste von Postfach F 53683, Herten 6

C-64-Forth. Diese neue Sprache jetzt auch f. **C-64** Cass.-Vers. DM 63,–/76,–, Diskvers. inkl. Handb. (einz. DM 25,–) **D-BASIC,** prof. BASIC erw., DM 33,– D. Luda, Staudinger Str. 65, 8000 München 83

Verk. TI-99/4A + Ex-Basic + Sprach-Synthesizer + Joyst.-Adapter + ca. 100 Prg. + Zubehör. VB. A. Neuburger, Feldstr. 16, 7505 Ettl. 8, Tel. 0 72 43 / 2 91 75.

TI 99/4 A Ext.-Spiel nur für Erwachsene – bitte Alter ang. zum absoluten Preishit von nur DM 10,- im Umschl. an T. Karbach, Remscheider Str. 18, 5650 Solingen 1

Wenn ihr „Mugsy von Melbourne House" wollt, dann kriegt ihr es für 35 DM (vielleicht) noch bei: Jan Weigner, Carl-Schurz-Str. 11, 2800 Bremen 1, Tel. 04 21 / 34 63 87

Wegen Systemwechsel zu verkaufen: **Laser 2001,** Diskettenlaufwerk, Datenrecorder, Printerkabel + Software auf Diskette. VB 1100,– Tel. 0 61 71 / 7 94 73 ab 18 Uhr

Computerstatik auf dem C 64
Anwenderfreundliche und übersichtliche Statikprogramme für das kleine und mittlere Architekturbüro als preiswerte Diskette bauaufsichtlich geprüft
Ing.-Büro S. Mergel, Zehntstraße 12, 3559 Rosenthal

★★★ **FLOPPY 1541 6 x SCHNELLER!** ★★★ durch Parallelbus für C 64. Interface kpl. geprüft mit Software nur DM 179,–. Info gegen DM 1,50 in Briefmark., B. Akesson, Pf. 802, 4040 Neuss.

Ankäufe

TI-99/4A
Suche **Extended Basic Modul** evtl. Tausch gegen **Editoriassembler** sowie Maschinensprachprogramme z. B. **Flugsimulator** und **Ex. Basic Spiel Miner 2049** (USA-Version).
Tel. 0 71 32 / 3 76 08

Suche
VC-20-Programme
Fritz, Wichdorfer Str.14
3501 Emstal 2

Suche **Drucker** für ZX81 (mögl. gebraucht). Tel. 0 26 41 / 33 62

Suche: Gute und billige Software auf Diskette für 64K Atari 600XL. Angebote an: Michael Ebeling, Riepener Straße 7, 3061 Beckedorf

TI 99/4 A, Zubehör, Module und Programme gesucht. Chiffre. ■

TI 99/4A Minimemory + Logo gesucht. Tel. 08 81 / 6 21 64

C-64. Suche Assembler, Pascal, Forth und andere Programmiersprachen. Tauschmaterial vorhanden. Stefan Schneider, Meerschlader Weg 1, 5275 Bergneustadt

Kontakte

Hallo Spektrumfreaks. Ich möchte mit euch gerne Prog. tauschen. Liste, ob groß oder klein, an: Robert Weidinger, Schönmetzlerstr. 10, 8050 Freising oder Sams.– Diens. 0 81 61 / 43 72

Kontakt mit MZ700-Leuten gesucht. Tel. 05 11 / 42 29 75

***** Atari-Spectrum-User-Club *****
monatl. Info, bundesweit organisiert, riesige Prog.-Bibliothek usw., melden bei:
ASUC, Am Schürhof 17
4905 Spenge
Tel. 0 52 25 / 20 68

Comp.-Club
Bad Hersfeld
sucht Mitglieder!
Kontakt über
Roland Reyer
Am Giegenberg 21
6438 Ludwigsau 1
oder Jugendhaus Hersfeld

Tausch

● ZX-Spectrum ● neue Programme ● Tauschbörse ● Info von S. Sauer ● Heideweg 8, 2720 Rotenburg ●

Tausche/Verkaufe Atari Prg. 16 K. Uwe Schöneberger, Sonnenstraße 18, 6652 Bexbach 2

Suche Software aller Art (Disk.), für Atari 800 XL u. Antic-Hefte bis März 1984, evtl. auch Computer u. Analog. Dietmar Levenhagen, Landwehr 83, 3057 Neustadt 1, Tel. 0 50 32 / 6 11 81

ZX-Spectrum
Software und Interessenaustausch
Monika Baumgartner
4021 Linz
Postfach 142 Austria
Bitte Liste beilegen

Auftrag für Gelegenheitsanzeigen in Homecomputer

Homecomputer
Landstraße 29
Postfach 41
3444 Wehretal 1

Unter der Rubrik „Kleinanzeigen" veröffentlichen wir Gelegenheitsanzeigen für Verkaufsangebote, Kauf- und Tauschgesuche, Kontaktaufnahme bzw. Erfahrungsaustausch usw.

Preise für „Kleinanzeigen":
Private Gelegenheitsanzeige je Druckzeile 5,— DM inkl. MwSt.
Chiffregebühr je Anzeige 10,— DM.
Gewerbliche Gelegenheitsanzeige je Druckzeile 11,— DM inkl. MwSt. (dürfen nicht unter Chiffre erscheinen).

Name und Adresse

Abo-Nr.

Unterschrift — Ich zahle sofort nach Rechnungserhalt — Datum

Ich wünsche folgenden Text zu veröffentlichen:

Zutreffendes ankreuzen

☐ Ankäufe ☐ Kontakte ☐ Geschäftsempfehlungen ☐ gewerblich
☐ Verkäufe ☐ Tausch ☐ Verschiedenes ☐ privat

Club-Ecke



Neugründung eines TI-Clubs

„Seit etwa einem halben Jahr gibt es bei uns in Münster einen TI-Club. Er besteht aus etwa 20 Mitgliedern und sucht weiteren Zuwachs.
Der Club entstand aus dem Bedürfnis heraus, Kontakt zu Gleichgesinnten aufzunehmen, Erfahrungen auszutauschen, Probleme zu bewältigen und den Umgang mit dem TI-99/4(A) effektiver zu gestalten.
Wir sind daher an einem regen Informationsaustausch mit allen TI-Anwendern interessiert. Von unserer Seite sind wir bereit, diesen Personenkreis mit Informationen aus dem Club zu versorgen und auch sonst nach unseren Möglichkeiten, Ratschläge zum Umgang mit dem TI-99/4(A) zu erteilen.
Wir planen, in regelmäßigen Abständen eine Clubzeitschrift herauszugeben, welche alle wichtigen Informationen über unseren Club und wesentliche Neuerungen enthalten wird.
Sofern Interesse an einem Kontakt zu unserem Club bzw. am Bezug unserer Clubzeitschrift bestehen sollte, bitten wir um schriftliche oder telefonische Kontaktaufnahme.
Wir möchten ausdrücklich darauf hinweisen, daß wir keinerlei kommerzielle Interessen verfolgen. Alle unsere Informationen und sonstigen Hilfen werden daher zum Selbstkostenpreis angeboten.
Die Mitglieder unseres Clubs wünschen sich insbesondere auch persönliche Kontakte zu Mitgliedern anderer Clubs."
c/o Thomas A. Schneider, Schillerstraße 65, Tel. 02 51 / 66 30 90

Sinclair User Club Schweiz

Wir suchen Mitglieder(innen). Ebenso wünschen wir Kontakt mit gleichgesinnten Clubs in Deutschland und Österreich.

Die Absicht dieses Vereins besteht darin, Erfahrungen, Tips und Tricks im Umgang mit Sinclair-Computern auszutauschen sowie in der Schweiz nicht erhältliche Hard- und Software zu vermitteln.

Interessierte Personen und Vereine melden sich bitte schriftlich oder telefonisch (jeweils abends) bei: Sinclair User Club Schweiz, Postfach 16, CH-8627 Grüningen, Tel. 01/2 41 19 08 oder Tel. 01/9 48 00 25

Korrektur

Maniac
für Atari
Seite 71

Ausgabe 1/85
Bitte überprüfen Sie nur die letzten Ziffern!

```
20505 PLOT 1,17:DRAWTO 7,17:PLOT 11,17:DRAWTO 15,17:PLOT 2,15:DRAWTO 7,15:PLOT1
2,15:DRAWTO 15,15
20510 COLOR. 163:PLOT 2,13:DRAWTO 8,13:PLOT 10,13:PLOT 11,13:PLOT 13,13:DRAWTO 16
,13:PLOT. 1,11:DRAWTO 8,11
20515 PLOT 13,11:DRAWTO 16,11:PLOT 3,9:DRAWTO 9,9:PLOT,13,9:DRAWTO 17,9:PLOT 7,7
:DRAWTO 17,7
20520 PLOT 8,5:DRAWTO 14,5
20525 COLOR 166:PLOT 6,22:PLOT 4,16:PLOT 2,14:PLOT 7,14:PLOT 2,12:PLOT 8,12:PLOT
12,22
20530 COLOR 4:PLOT 2,22:PLOT 2,18:PLOT 3,20:PLOT 3,16:PLOT 3,12:PLOT 3,10:PLOT 4
,14:PLOT. 5,18:PLOT 9,20
20535 PLOT 11,22:PLOT 11,18:PLOT 11,6:PLOT 13,16:PLOT 13,12:PLOT 13,8:PLOT 14,14
```

Electron

Der neue Home-Computer von Acorn Computers

Für den Preis eines Home Computers bietet er Ihnen viele Eigenschaften, die Sie sonst nur bei professionellen Micros im Ausbildungsbereich oder im Geschäftsleben finden.

Acorn Basic

Der Electron ist die ideale Maschine, mit der Sie ins Computern einsteigen können – mit speziellen Lernprogrammen oder auch mit Spielen. Seine graphischen Möglichkeiten gehören zum Besten, was es in dieser Preisklasse auf dem Markt gibt.

Das vielgelobte ACORN BASIC ist eingebaut. Fachleute bezeichnen es als eine besonders leistungsfähige und komfortable Programmiersprache. Mit ACORN BASIC entwickeln Sie von Anfang an einen guten und rationellen Programmierstil, der Sie immer begleiten wird.

ACORN BASIC ist die Sprache des in Großbritannien überaus populären British Broadcasting Computers, der ebenfalls aus dem Hause Acorn kommt und auch in Deutschland erhältlich ist. In Großbritannien hat dieser Computer eine eigene Fernsehserie.

Hochwertige Ausstattung

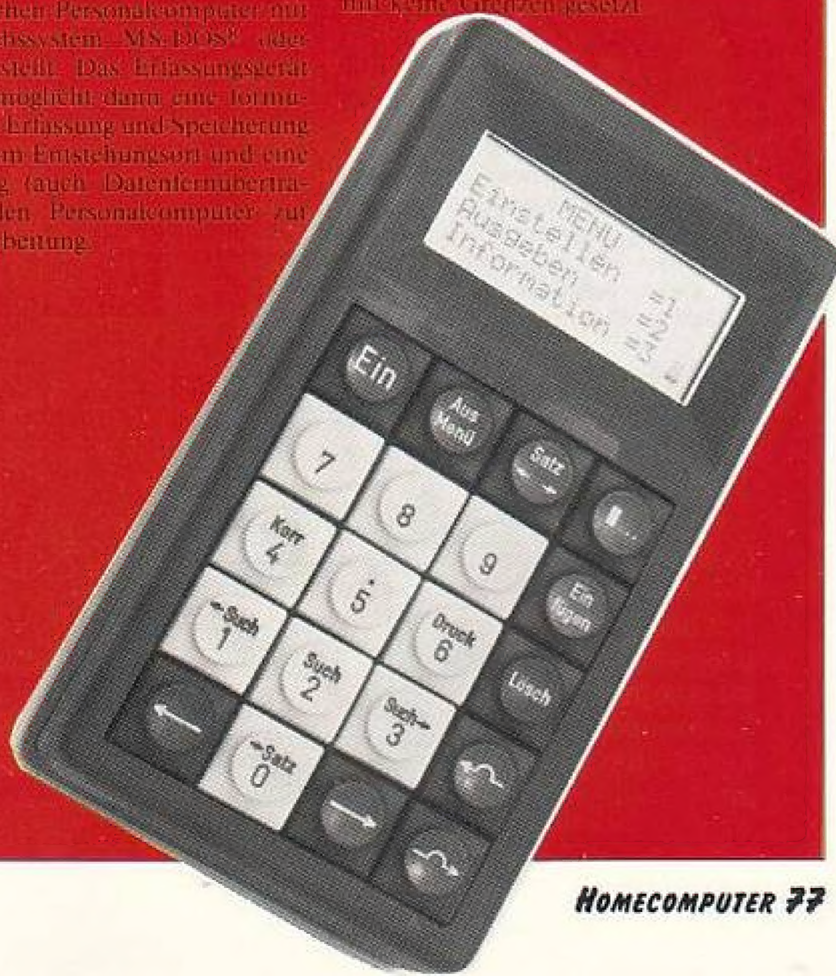
Der Electron hat ein stoßfestes, ergonomisch gestaltetes Gehäuse. Die Schreibmaschinen-Tastatur besitzt 56 Tasten, von denen Sie zehn nach eigenen Bedürfnissen frei definieren können. Zahlreiche Steuerbefehle lassen sich durch einen einzigen Tastendruck abrufen – eine große Hilfe beim Programmieren (shorthand BASIC).
Auf einem Monitor oder einem Farbfernsehgerät stellen Sie die hochauflösende Graphik dar. Zum Speichern und Laden von Programmen können Sie jeden handelsüblichen Cassettenrecorder anschließen. Ein Diskettenlaufwerk wird schon bald auf den Markt kommen.

Der Electron besitzt eine Speicherkapazität von 64 K. Davon entfallen 32 K auf den Lesespeicher (ROM) mit dem Betriebssystem und dem BASIC-Interpreter. Der Arbeitsspeicher (RAM) umfaßt 32 K, von denen 3,5 K für das Betriebssystem benötigt werden.

Datenerfassung leicht gemacht

Unter diesem Motto ist das neue Datenerfassungsgerät ME 820 der GSI Datentechnik GmbH in München zu sehen. Nicht größer als ein Taschenrechner, aber mit einer beeindruckenden Leistungsbreite.
Die individuellen Erfassungsformulare (Masken) werden im Dialog auf einem herkömmlichen Personalcomputer mit dem Betriebssystem MS-DOS® oder CP/M™ erstellt. Das Erfassungsgerät ME 820 ermöglicht dann eine formularbezogene Erfassung und Speicherung der Daten am Entstehungsort und eine Übertragung (auch Datenfernübertragung) an den Personalcomputer zur Weiterverarbeitung.

Das mobile Datenerfassungsgerät ME 820 ist 500 g leicht, besitzt eine alphanumerische LCD-Anzeige mit 4 Zeilen à 16 Zeichen und eine Benutzerführung über Bildschirmformulare sowie einen akustischen Signalgeber und einen 16-K-Byte-Programmspeicher. Der Datenspeicher reicht von 8 bis 48 K-Byte. Eine RS 232C-Schnittstelle ermöglicht den Anschluß an den Personalcomputer mit Modems, Akustikkoppler oder direkt. Ein umfangreiches Software-Paket wird mitgeliefert und ist bereits im Preis enthalten. Der individuellen Nutzung, zum Beispiel Auftrags-, Bestell-, Produktions-, Verbrauchsdaten-Erfassung etc., sind somit keine Grenzen gesetzt.



Postkarte einsenden an:

Tronic-Verlag, Postfach 34 44 Wehretal 1

 oder einfach telef. bestellen: 0 56 51 / 4 06 43-4 06 93

Spiders:



Ziel des Spieles ist es, zwei Spinnen, die auf ihren Netzen herumkrabbeln, zu beseitigen. Vernichten kann man die Tierchen allerdings nur mit Insektengift. Davon liegt noch genug im Keller, aber wo ist der Schlüssel? – Unser Topprogramm aus der Ausgabe Oktober. Spiders gefällt durch eine gute Grafik und guten Sound. Gespielt wird mit Joystick an Port 2. **Für Commodore 64.**

Frogger:



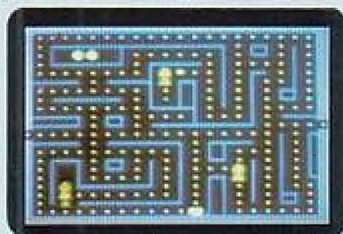
Ein Spiel für den **ZX-Spectrum 48K.** Die beliebte Spielversion jetzt auch für den ZX-Spectrum. Ein Frosch hockt am Straßenrand einer viel befahrenen Straße und versucht verzweifelt, sie zu überqueren. Ist ihm das gelungen, wartet eine zweite schwere Aufgabe auf ihn. Ein Fluß, verseucht mit Krokodilen, muß überquert werden. Das Spiel verfügt über einen guten Sound.

Mother Duck:



Ähnlich der Spielversion „Frogger". Eine Entenmutter muß einen Fluß überqueren, um Futter für ihre Jungen zu bekommen. Allerlei Untier hält sich jedoch im Wasser auf und macht jede Flußüberquerung zu einer gefahrvollen Angelegenheit. **Für TI 99.** Benötigt werden Joystick und das **TI.-Ext.-Basic-Modul.**

Fressmann:



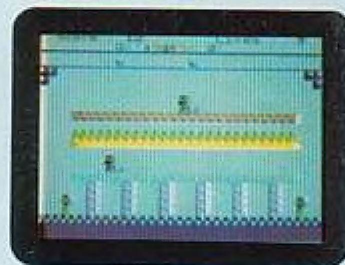
Fressmann ist eine Pacman-Version. Das Programm, welches **ohne Erweiterung** lauffähig ist, steht der Originalversion in nichts nach. Fressmann läuft auf dem **VC 20** und wird mit Joystick gespielt.

Projekt:



Als Topprogramm auserwählt von der Redaktion. Sie sollen eine gefährliche Mission ausführen. Um für die nächste Zeit genügend Uran zur Verfügung zu haben, sollen Sie eine Reise zum Mond unternehmen und nach dem edlen Metall suchen. Ihre Reise ist in fünf Phasen aufgegliedert, die nacheinander bewältigt werden müssen. Ein schönes Spiel für den **Commodore 64.**

Oil-Panic:



Sie müssen versuchen, möglichst viele Tropfen aufzufangen, die aus einer defekten Ölleitung heraustreten. Aber aufgepaßt, denn der Eimer, der zum Auffangen da ist, ist mit nur vier Tropfen gefüllt. **Für ZX-Spectrum 48K.**

Skeet: (Tontaubenschießen)



Für **Commodore 64.** Dem realistischen Tontaubenschießen nachempfunden. Von einem Katapult geschleuderte Tontauben müssen reaktionsschnell getroffen werden. In diesem Spiel kann jeder seine eigene Meisterschaft austragen. Eine Supergrafik zeichnet besonders aus.

High Noon:



Ein tolles Spiel für den **Commodore 64.** High Noon besitzt eine schöne Grafik und eine hohe Spielgeschwindigkeit. Gewählt werden kann zwischen 255!! verschiedenen Spielstufen. – Auf einer vielbefahrenen Postkutschenstraße stehen sich zwei Cowboys gegenüber und duellieren sich. Ein Spiel für alle Western-Freunde.


Das muß nicht sein ...
... auf diese Art und Weise gute SOFT-WARE zu bekommen.
Nutzen Sie doch einfach unseren preis-günstigen Katalog.
Mehr darüber auf den Seiten 78 und 79.